1992 / SZEPTEMBER

ÁRA: 196 FT

# ALAPLAP

MIKROSZÁMÍTÓGÉP MAGAZIN MÁGNESLEMEZ MELLÉKLETTEL




A HÓNAP TÉMÁJA:

## TUDOMÁNYOSODUNK

A MÁGNESLEMEZEN:

Szövegdaraboló
Szimulált multitaszk
CEB-vírusok ellenszere
Szorzótábla-gyakorlat
Gumidominó

Minő menő menük!

Az EXPO kockázata

Las Vegas csak egy ugrás!

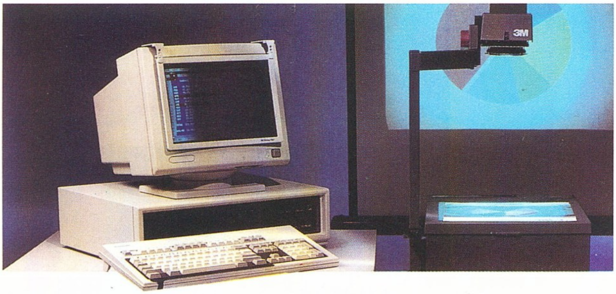
Gyónnak a szoftverkalózok

Fogd a mágneskártyád, és légy vidám!

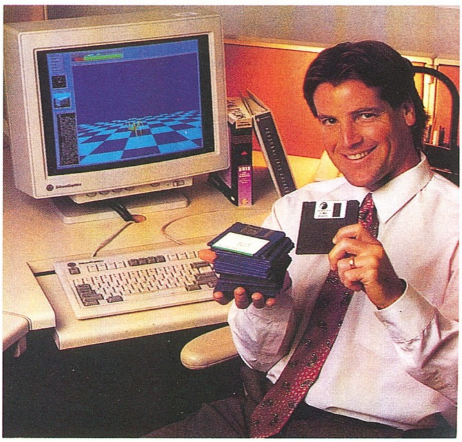
Milyen VGA rendszert válasszunk?

Hogyan ismerhetjük fel a vírusokat?

Compfair '92: október 6–10.


3M





ELIN

*ELIN ELEKTRONIKA BUDAPEST KFT*
*1072 Budapest*
*Dob u. 54.*
*Tel.: 142-3734*
*Fax: 122-6423*

*Az 1892-ben alapított osztrák ELIN magyarországi leányvállalatának célja, hogy magas színvonalon elégítse ki a hírközlés, az információtovábbítás iránti egyre növekvő igényt, a legkorszerűbb berendezések széles választékát kínálva a szokványos berendezésektől a vevő különleges igényeit kielégítőkig.*

*A kínálat a hordozható rádióadó-vevőktől a vezeték nélküli távvezérlő rendszeren át, a legkorszerűbb hírközlő rendszereket, vezeték nélküli telefonokat, mikroprocesszor-vezérlésű rádió-távirányítású rendszereket, elektronikus közlekedésbiztonsági berendezéseket és meteorológiai rendszereket nyújtja.*




INFORMÁCIÓKÉRÉS: 02 ▶

# ALAPLAP

Mikroszámítógép magazin mágneslemez melléklettel

Megjelenik havonta

Főszerkesztő: Faklen Pál

Főszerkesztő-helyettes: Varga János

Szerkesztő: Jakab Ágnes

Munkatárs: Sziebig Andrea

A Lemezkalauz és a Közkincs szerkesztője: Verebély Pálné

A szerkesztőbizottság tagjai: Barna László, Boros György, Broczkó Péter, Brüll Károly, Farkas Ernő, Feleki Zoltán, Herczeg József, Kassay Árpád, Kónya László, Kovács P. Attila, Pintér Gábor, Vargha Dénes, Vékony Tamás, Villányi László, Zoltai Péter

Szerkesztőség, kiadó és hirdetésszervezés:

1441 Budapest
VIII., Reguly Antal u. 8.
Telefon és fax: 133-1839

Felelős kiadó: Sebestyén Ilona ügyvezető igazgató

Cédrus Kiadó Kft

Nyomdai előkészítés: Tipoprint Kft, Budapest

Nyomtatás: Zalai Nyomda, Zalaegerszeg
Felelős vezető: Galla József

Terjeszti a Magyar Posta. Előfizethető a hírlapkézbesítő postahivataloknál és a Posta Hírlapelőfizetési és Lapellátási Irodájánál (XIII., Lehel u. 10/a, Budapest 1900), vagy átutalással a 215-96162 pénzforgalmi számra.

Példányonkénti ár: 196 Ft
Évi előfizetési díj: 2 352 Ft

Külföldre terjeszti a Kultúra, Pf. 149, Budapest 1389

HU ISSN 0865-9788



## A HÓNAP TÉMÁJA: TUDOMÁNYOSODUNK

4 „Gépidő” (Kálmán János)
6 A gépi bizonyítás értéke (Pogány Csaba)
7 A poli(p)technikai oktatás (Szondi Egon János) ❑
9 A gazda szeme (Ferenczy Tibor—Tóth József)
10 Gyógyszermodellezés (Bordás Barna)
14 A medicina „szívében” (Naszlady Attila)
16 Ami a térképen nem látható (Kováts Zsolt)
21 A „pokoli tornyok” megfékezése (Tilly Károly)
23 Diagnózis egyszerű és „cifra” esetekre (Somogyi Péter)

## SZOFTVERTÉKA

25 Test test ellen? (Herczeg József)

## KIRAKAT

29 Las Vegas csak egy ugrás! (Faklen Pál)

## KÖZKINCS

30 Minő menő menük! (Verebély Pálné)
31 Sétálunk... az aknamezőn (Lampert Csilla)
32 Windows-újdonságok — kimazsolázva (Verebély Pálné)
33 Logika és lövöldözés (Gerlits Judit)
34 Őszi SolarSoft újdonságok (Verebély Pálné)
34 Hangsztár zenés napokra (Verebély Pálné)

## TUDÁSTECHNOLÓGIA

37 Szakértőrendszer-projektek és -alkalmazások (Sántáné Tóth Edit)
38 MProlog-alapon (Farkas Zsuzsa)
40 Az EXPO kockázata (Futó Iván)
42 Turbógenerátor-rezgésdiagnosztika (Bessenyei Zoltán—Tomcsányi Tamás)

## SOLARSOFT LEMEZKALAUZ

## FOGÓDZÓ

45 Az információ mozgatása (Csórián Sándor)
47 Fogd a mágneskártyád, és légy vidám! (Starcz Andor)
50 „Tesztkontroll” (Herczeg József)

## HÁLÓZAT

53 Hírzárlat (Villányi László)

## ALAPJÁRAT

54 A szokott helyen és időben (Sziebig Andrea)

## 56 BÖNGÉSZDE

## SZÖVEGELŐ

57 Ne kenjük a gépre! (Faklen Pál)

## SZERSZÁMOSLÁDA

59 A „klasszikus” Mathematica (Kónya László)

## GÉPRAJZ

62 Kényelmesen, profi módon (Lóth Tamás)

## VÍRUSŐRJÁRAT

67 Egy terjedési mechanizmus tanulságai (Hornák Zoltán) ❑
68 Hogyan ismerhetjük fel a vírusokat? (Rákos Péter)
69 Újdonságok McAfee 93-as csomagjában (Nagy Gábor)
70 Ami a Vírushatározóból kimaradt...

## 71 MIKROBAZÁR

## 71 KÖNYVESPOLC

## KALEIDOSZKÓP

73 Sziget a nyelvek óceánjában (Vargha Dénes)

## KILÁTÓ

77 Milyen VGA rendszert válasszunk?
78 Gyónnak a szoftverkalózok

## PROGRAMOZÁSTECHNIKA

83 Új sorozatunk (Fridl György)
84 Még mindig olvasunk... (Nemes Mihály)
86 Csak könnyedén, könnyeden... (Vargha Dénes) ❑

## PALETTA

90 Hardver-szoftver — innen-onnan (Sziebig Andrea)

## MÁGNESLEMEZ MELLÉKLET ❑

**Feleki Zoltán karikatúrái**

51 E számunk hirdetői

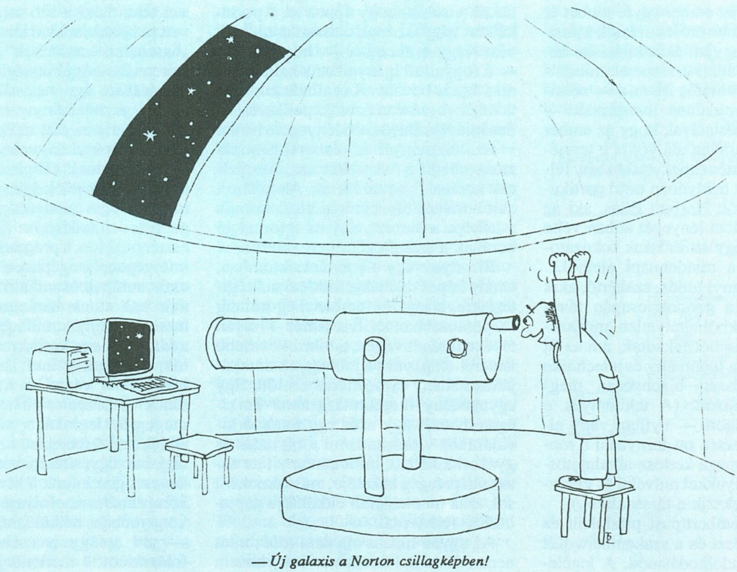
# Tudományosodunk

Ne ijedjenek meg az olvasók, kedvenc (alap)lapjuk nem kíván sem a filozófia, sem a pszichológia, sem a kémia, sem egyéb exkluzív szaktudományok orgánumává válni. Nem szeretnénk olyan irányban sem manipulálni senkit, hogy rokonszenvessé tegyük a „tudományos" arisztokratizmust, az elméletieskedést, a misztifikáló stílust, amely néha már-már lefordíttatásra szorul. Sőt, azt szeretnénk, ha mindez a számítástechnikát nagy ívben elkerülné!

De az már régóta a „levegőben lógott", hogy a hónap témája egyszer a számítástechnikának egyes tudományos szakterületeken való alkalmazása legyen, mert a számítógép szép csendben meghódította olyan tudományágak ormait is, amelyek sokáig nem sok esélyt adtak a „buta" gép bevonulásának. És ez már nem a légkondicionált, nagytermes — bár a térfogattal nem mindig arányos kapacitású — számítóközpontok világa. A PC a tudományban is megjelent „minden kilométerkőnél" — pontosabban majdnem minden íróasztalon.

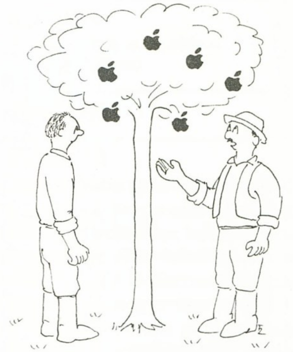
Témaválasztásunk másik oka, hogy az információáramlás eddigi egyoldalúságán is változtatni szeretnénk. Lapunkban — a dolog természetéből fakadóan — többnyire arról esik szó, hogy mit tud nyújtani a számítástechnika az egyes alkalmazási területeken — hardverrel, szoftverrel..., vagy hogy mit kell valakinek megtanulnia, hogy a gépeket, programokat használni tudja. A „pult" túlsó oldalával viszonylag ritkábban foglalkozunk: azzal, hogy a hivatásos számítástechnikusok ismerjék meg jobban a tudományos és szakmai területek lehetőségeit, igényeit, sajátosságait, ötleteit, kritikai megnyilatkozásait... Meggyőződésünk, hogy a gyakorlatban igazán eredményes számítástechnikai megoldásokat az adott szakterület nagyon alapos ismerete, vagy egy nagyon mély és kreatív párbeszéd előzi meg.

A tudományok számítástechnikai eszköztárának igen jót tenne, ha a bitvadászok gyakrabban vetnék tekintetüket a monitoron túli valóságot feltárni igyekvő diszciplínákra is. Hátha vannak ott érdekes és új dolgok számukra...



— *Új galaxis a Norton csillagképben!*

# A receptekből összetákolt világ
# „Gépidő"



– Ilyen a termés, mióta számítógépet használok

„Where is life, we lost in living;
where is wisdom, we lost in knowledge;
where is knowledge, we lost in information?"
(T. S. Eliot)

Prózában és magyarul:
„Hol van az élet, ami elveszett a létezésből;
hol a bölcsesség, ami kiveszett a tudásból;
hol van a tudás, amit elnyelt az információ?"

A második világháború utáni „arany évtizedek" felhőtlen optimizmust hoztak. Világszerte eluralkodott egy derék, erkölcsnemesítő, pozitivista elképzelés — meggyőződés, hogy a haladás folyamata nem más, mint az emberiség fokozatos kiemelkedése a történelem előtti idők barbárságából. Nem csoda, hiszen ezt az álláspontot már előzőleg is olyan tekintélyek támogatták, mint a francia és a brit történetírás liberális óriásai, akik a történelmet a ráció és a babonaság küzdelmeként ábrázolva, az Ész fokozatos diadalára esküdtek: Max Weber a racionalizálódás-elméletével, amely szerint az ember egyre többet ér el, mind kevesebb erőfeszítéssel; a pszichoanalízis nagyjai, akik kilátásba helyezték, hogy állati ösztöneink mindinkább megfékezhetők; Marx a — némileg talán egyoldalúan interpretált — nagyszabású jóslatával, hogy az ember fokozatosan uralma alá vonja a természetet és a történelmet, miközben felszabadul az őt béklyóban tartó partikularizmusok alól; Norbert Elias, aki az újkori történelem lényegét abban vélte fölismerni, hogy az erőszak fokozatosan kiszorul a mindennapi életből... No, meg seregnyi tudós, szakértő, akik kezeskedtek: a gondok csupán téves megállapításokból, helytelen módszerek alkalmazásából fakadtak, ám az új felismerések, a tudomány és a technika fejlődése egészen bizonyosan megszünteti a bajokat. (A tekintélyek e koalíciója mögött — nyíltan vagy elrejtőzve — persze ott tornyosul a modern állam a maga kertész-alkalmatosságaival, amelyekkel művelni és gyomtalanítani igyekszik a társadalmat.)

Két tiszta embertípust produkált ez a kor: a művészt és a szakembert. Két alkat, két gondolkodásmód. A legélesebb határozottsággal zárják ki egymást, s mégis, a megkülönböztethetetlenségig hasonlatosak. Mindkettőnél öncéllá lettek az eszközök, amelyeknek értelmet és jelentőséget csak célra irányulásuk adhatna. Mindkettő az ember egy-egy „szeletét" nézi, az egész egyéniséget nem érinti. Mélyen szakmaszerű lett a művészet: a zenészek zenészeket, az irodalmárok irodalmárokat, a képzőművészek képzőművészeket próbálnak elkápráztatni legfrissebb technikai bravúrjaikkal; ezzel egy időben (a mosóporgyártókhoz hasonlóan) igyekeznek meggyőzni a vastagabb pénztárcák viselőit, hogy éppen az ő portékáikért vágyakoznak olthatatlanul. Művész vagy szakember — ha élni akar — a fogyasztó igényeihez alkalmazkodik; és ez, bevallva-bevallatlanul, nem is esik nehezére: a fizetőképes keresletben mindinkább önnön lényegére ismer.

Az eszmegyártás és -reklámozás szakemberei a korszakot az „információ korának" nevezték el. Ahogyan a „hit korának", az európai középkornak a jelképe a kereszt, úgy az információ korának a számítógép.

Bizonyos: egy olyan társadalomban, amely képes rakétákat küldeni más égitestekre, a legtöbb ember még mindig — pontosabban szólva, ismét — olyan tevékenységet végez, amihez kevesebb ismeret elég, mint a jobbágyok munkájához néhány évszázaddal ezelőtt. Egy agyagedény megformázásához és kiégetéséhez, vagy akár egy fateknő kivájásához vitathatatlanul több tudás és gyakorlat kellett, mint amennyit az automata présgép kezelője, műszakonként sok száz munkadarab előállítója napjainkban felhasznál.

Az egyén tudása ma nem *több*, mint nemzedékekkel korábban volt, hanem *más*. Ami alapvetően megváltozott: az össztársadalmi tudás struktúrája.

Az egyén munkája — munkatevékenységének szűkülésével — mind sokrétűbben kapcsolódik másokéhoz, és ez mind kevésbé valósulhat meg személyes kapcsolataiban. Ezek helyébe a többiek tárgyiasult tudásának elsajátítása lép. A közvetlen tapasztalatokon nyugvó és személyes érintkezéssel, közös tevékenység útján hagyományozódó ismereteket kiszorítja a készen átvett, absztrakt jelek (írás, képletek, ábrák) közvetítésével szerzett tudás. Ez a látszat ellenére csak kivételesen tekinthető a szó teljes értelmében vett tudásnak, sokkal inkább a különféle „használati utasítások", know-how-k ismeretének és jártasságnak ezek alkalmazásában. Így használja a gépszerkesztő a kézikönyveit, az orvos a gyógyszeripari cég tájékoztatóit, a jogász a törvénykönyveket.

Amit leírtunk: féligazság. Nem pusztán absztrakt jelek közvetítésével szerezzük meg a mindennapjainkhoz elengedhetetlen tudást. Az esztergálás csak esztergapadon, a programozás csak számítógépen, a gépkocsivezetés pedig csak autóban tanulható meg. Amikor már hallottunk harangozni valamit a motorról, sejtjük, mi a gyújtás, a fék és a kuplung, esetleg a kresszt is bebifláztuk, e magas szintű „elméleti tudás" birtokában végre a kormány mögé ülünk. Miközben — a hol lefulladó, hol megugró, leginkább a bakkecskéére emlékeztető mozgású autóban — végtagjaink ügyesítésén izzadunk, bizony keveset gondolunk a hengerben lezajló komprimálási folyamatra. Sokkal könnyebben nélkülöznénk oktatónkat — aki amúgy is csak idegességünk fokozására ül mellettünk, s akinek út-

mutatására a kritikus pillanatokban különben is süketek vagyunk —, mint a gépkocsit, „aki" végül maga tanít meg a bánásmódjára.

Világunk: receptekből összetákolt világ. Sámánja, varázslója, papja: a szakember, aki egy-egy receptkollekciót megért és alkalmazni tud. „Élő" tanítómestereink: a gépek. Az ismeretátadásnak ez a modern korra jellemző, sajátos közvetítettsége szüli köznapi vélekedésünket, miszerint a fejlődés motorja a technika, fő tényezői a modernebb és modernebb gépek.

A gépek... Ahogy Petőfinek vagy Walt Whitmannak a gőzgép, a vasút a tulajdonképpeni gép, gondolkodásunkban a modern gép par excellence számítógépként tűnik fel. „Az ipari gépek megszabadították az embert a legnehezebb, legfárasztóbb, elnyomorító fizikai munka terhétől, a komputerek pedig felszabadítják majd az unalmas, ismétlődő szellemi rutintevékenységtől, és alkalmassá teszik az igazi emberi alkotómunkára" — mondogatják a szakértők „A számítástechnika és a jövő", „Mikroelektronika — áldás vagy átok?" és hasonló címekkel nap mint nap megrendezett konferenciáikon. Igaz, az aggályoskodóbbak „bátran" rámutatnak a várható bajokra: nő a totális ellenőrzés lehetősége, nő az automatizálás és a munkanélküliség összefüggése stb. Ám abban erős az egyetértés, hogy a helyzet gondos felmérésével, helyes törvények, politikai és gazdasági intézkedések meghozatalával mindezek a problémák minimalizálhatók; a szabad piacgazdaságon alapuló modern állam a maga eszközeivel az új instrumentumokat kétségkívül a közjó szolgálatába állítja. A bizakodás természetes — de csak egy olyan gondolkodásmód számára, amely immunis az éppen fennállón kívül bármilyen racionalitással szemben, és ezért nem hajlandó, de talán képtelen is felfogni, mi és miért történik.

Az a történelmi kontinuum, ami összeköti a technika előtti, premodern és a mai társadalmakat: az emberek uralma emberek fölött. Mégis, az ipari technológia megváltoztatja az uralom alapját: a személyes függőségek (a rabszolga viszonya urához, a jobbágynak a földesúrhoz, a földesúrnak a hűbérúrhoz stb.) mintegy természetadta rendszerét fokozatosan „a dolgok objektív rendjéből" (a piaci viszonyokból, a gazdaság törvényeiből stb.) eredő „természetes" függőségek rendszere váltja föl. Bár „a dolgok objektív rendje" maga is az uralom eredménye, tény, hogy a korábbiaktól gyökeresen különböző racionalitást hoz létre: olyan társadalom racionalitását, amely — miközben fenntartja a maga hierarchikus struktúráját — növekvő mértékben aknázza ki a természeti és emberi erőforrásokat. Ennek jegyében az egyént mindinkább fegyelmezi, felhasználja és ellenőrzi egy olyan termelő apparátus, amely állandósítja, globalizálja és totálissá tágítja a létért vívott harcot — úgy, hogy közben tönkremegy az apparátust kiépítő, annak részét képező és azt felhasználó egyének élete.

Nyilvánvaló: magának a rendszernek a racionalitása az, ami katasztrofális. Egyre „tudományosabbá" és „racionálisabbá" válik a létért való küzdelem, azaz a társadalom a dolgok és viszonyok gyarapodó technikai együttesében reprodukálja önmagát — ez szükségszerűen foglalja magában az emberek technikai-technológiai hasznosítását. Egy történész, Henry L. Feingold így ír „How Unique is the Holocaust?" c. könyvében (1983): „Auschwitz ugyanakkor a modern nagyipari termelés kiterjesztése is volt. Termékeit, igaz, nem bocsátotta piacra, de alapanyagokat (emberi lényeket) használt fel, amit a korszerű európai vasúti hálózat szállított a hasznosítás helyére, és végtermékét, ennyi és ennyi ember halálát termelési grafikonokban regisztrálták. ... A szisztéma a modern tudományból nőtt ki, és olyan social engineering sémán alapult, amelynek hatékonyságát bármely fejletlenebb társadalom megirigyelhette volna." Raul Hilberg szerint „figyelembe kell venni, hogy a népirtás legtöbb részese nem nyitott tüzet kisgyermekekre, nem szórt mérgesgázkristályokat a gázkamrákba. A legtöbb elkövető csupán feljegyzéseket készített, előadmányokat fogalmazott, telefonbeszélgetéseket bonyolított, értekezleteken vett részt. Egy egész nép elpusztításában működhettek közre úgy, hogy közben fel sem keltek az íróasztaluk mellől."

A „tudományosan szervezett, racionális" munkamegosztás a személyes döntésből fakadó erkölcsi felelősséget háttérbe szorítja. E tényt nemcsak történelmi krízishelyzetekben, hanem köznapi életünkben is tapasztaljuk. (Emlékezzünk Konrád György „A látogató" c. művének gyámügyi előadójára, aki ügyfelét nem aktának tekintve, maga is emberséggel, s nem a bürokrata-morál normái szerint próbál segítségére lenni. Az eredmény végzetes.) A számítógépes nyilvántartás általánossá válása különösen elgondolkodtatóvá teszi, hogy:

*„Számon tarthatják, mit telefonoztam*
*s mikor, miért, kinek.*
*Aktákba írják, miről álmodoztam,*
*s azt is, ki érti meg.*
*És nem sejthetem, mikor lesz elég ok,*
*előkotorni azt a kartotékot,*
*mely jogom sérti meg."*

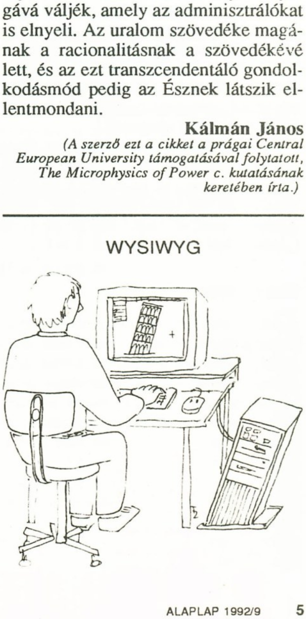
A modern polgári „józanság", tudományos munkamegosztás és szakigazgatás soha nem képzelt mértékben emelte a gazdasági, politikai és kulturális vállalkozás produktivitását. Egyidejűleg kitermel egy olyan filozófiát, amely létrehozza és igazolja a „racionális vállalkozás" nyomasztó, pusztító következményeit. A tudományos-technikai racionalitás (és manipuláció) a hatalomgyakorlás technológiájának döntő elemévé vált. A számítógép nem közömbös eszköz; amikor a komputerizáció a termelés általános formájává lesz, egy teljes kultúrát határol körül, egy történeti totalitást, egy „világot" konstituál. A tudományos gondolkodás és alkalmazása, a tudományos nyelv univerzuma és a mindennapi nyelv és viselkedés között szoros az összefüggés. Mindkettő az uralom logikájának és racionalitásának a jegyében mozog. A tudomány és a technológia az uralom leghatékonyabb hordozójává lett.

Nemcsak az egyén társadalmi helyzetét és másokhoz való viszonyát tünteti fel objektív törvények által meghatározottnak, hanem e törvényeket a racionalitás örök megnyilvánulásaiként jelenítik meg. A világ arra tart, hogy olyan totális adminisztráció nyersanyagává váljék, amely az adminisztrálókat is elnyeli. Az uralom szövedéke magának a racionalitásnak a szövedékévé lett, és az ezt transzcendentáló gondolkodásmód pedig az Északnak látszik ellentmondani.

**Kálmán János**

*(A szerző ezt a cikket a prágai Central European University támogatásával folytatott, The Microphysics of Power c. kutatásának keretében írta.)*

WYSIWYG



## Quod erat demonstrandum

# A gépi bizonyítás értéke

Lehet számítógéppel keresni egy adatot, egy cikk szerzőjét, de lehet például új tudományos igazságokat, új műszaki, sőt számítástechnikai konstrukciókat is. Matematikai kijelentések (ún. tételek) igazságának bizonyítása szintén beletartozik ebbe a körbe. A gépi bizonyításokra nem mindig a szellemesség, a szépség, hanem legtöbbször a kitartó munka a jellemző. Ez pedig hasznos, de nem szép. Az esztétikum és a praktikum nem mindig találkozik, még a tudományban sem, de nemcsak az esztétikus értékes, hanem a praktikus is.

Mivel semmilyen gép nem tekinthető abszolút hibátlan működésűnek, fel kell vetni azt a kérdést, hogy mit ér a gép munkája? Mit ér a gép által bizonyított tudományos igazság? De előbb: mi is a gépi bizonyítás?

A gépi bizonyítás semmi más, mint valamilyen következtetési folyamat gépi megvalósítása, leggyakrabban utánzása. Mivel a következtetéseknek a birodalma rendkívül gazdag, a gépi tételbizonyítások is nagyon változatosak. Három gyakori és tipikus csoportot azonban érdemes kiemelni. Az első az absztrakt területek absztrakt tételeinek bizonyítása, axiomatikus szigorral, logikai műveletekkel. A második: konkrét kijelentések igazságának a vizsgálata, illetve eldöntése. A harmadik a kettő között áll, miszerint egyikből is, másikból is tartalmaz elemeket.

Előfordulhat, hogy valamilyen vizsgált kijelentésről, ami igaz, azt állapítja meg a gép, hogy az nem igaz („elsőfajú hiba"), de az sem zárható ki, hogy igaznak minősíti ezt, holott nem az („másodfajú hiba").

### Hiba-tudomány

A hibák okainak, természetének, gyakoriságaiknak és egymással való összefüggéseiknek a vizsgálata rendkívül érdekes tudomány, nem kevésbé érdekes és hasznos természetesen a hibák megelőzésének, megkeresésének és javításának mestersége vagy inkább művészete.

A hibák világának rengeteg érdekes problémája közül kiemelünk egy olyan kérdést, amely jól szemlélteti, hogy ez a fontos terület mennyire nehéz terep is egyben.

Vegyünk például egy hibakereső vagy egy hibajavító eljárást! Ha a hibakeresést/hibajavítást gép, tehát olyan eszköz hajtja végre, amely nem tekinthető abszolút hibátlan működésűnek, előfordulhat, hogy a gép bizonyos hibákat megtalál, illetve kijavít, másokat nem, sőt új hibákat is bevihet. Mit tegyünk hát?

Jelenleg sok ilyen és ehhez hasonló, a gyakorlatban nagyon fontos kérdésre nem tudunk megnyugtatóan válaszolni. Tehetetlenségüket, tudatlanságukat háromféleképp szokták kezelni az emberek. Vagy bevallják, vagy nem vesznek tudomást róla, vagy álmegoldásokkal védekeznek. Korunk számítástechnikája a megbízhatósági, pontossági kérdésekről általában nemigen vesz tudomást. Néhányan azonban tudományos álmegoldásokkal nyugtatgatják magukat és másokat. (Nehéz megállapítani, hogy melyik a rosszabb.)

Vannak tudományos hiedelmek, amelyek nem egészen tökéletes alkatrészekből korlátlan tökéletességű berendezések létrehozását hiszik megvalósíthatónak. Mivel a hit magánügy, ezzel a kérdéssel nem is kellene többet foglalkoznunk, sajnos azonban a „nem egészen tökéletes alkatrészekből korlátlan tökéletességű berendezés létrehozhatóságának tételét" egyesek bizonyítottnak veszik, és széltében-hosszában hirdetik is. Ez a csacsiság azonban nem sokkal ér többet, mint az „örökmozgó", hasonlít is hozzá, hiszen ezt is újból és újból elővészik, sőt bizonyításokat is kreálnak hozzá.

Ezek a kérdések elméleti és gyakorlati szempontból egyaránt fontosak, törődni is kell velük, de most mégsem azzal foglalkozunk, hogy ha a gép igaznak vagy hamisnak talál valamit, akkor az mindig, és tényleg úgy igaz-e, hanem a módszerekre vetünk egy futó pillantást.

### Becslések gépi vizsgálata

A gyakorlatban ritkán kapunk munkánkhoz pontos adatokat, de nincs is szükség sokszor ilyenekre, hiszen ritkán kell pontos eredményeket szolgáltatnunk. A számolási munkákban jobbára a legtöbb adat becsült adat. Fontos kérdés, hogy meghatározott pontosságú kiinduló adatokból (ha ez egyáltalán lehetséges) hogyan lehet e kiinduló adatok valamilyen adott függvényének egy előírt pontosságon belül maradó közelítését a leggyorsabban kiszámítani.

Tehát nem szükségképpen az adott függvényt kell kiszámítani, hanem egy olyan függvényt, amely ennek a függvénynek valamilyen környezetében, tehát egy bizonyos tartományban van.

A feladat az, hogy keressük az egy tartományba eső (valamilyen tulajdonságú) függvények közül a leggyorsabban kiszámíthatót. A keresett függvénynek, illetve függvényeknek a tartomány határait jelző függvények között kell lennie. Tehát a felső határ (vagy korlát) alatt és az alsó felett.

Hogyan ellenőrizhető az, hogy egy függvény megfelel-e az ilyen követelményeknek? A megfelelés eldöntéséhez függvények közötti egyenlőtlenségeket, tehát becsléseket kell bizonyítani vagy cáfolni. Ezt megtehetjük elméleti eszközökkel is, papírral és ceruzával, elvégeztethetjük géppel, de elméleti területen maradva, viszont az is járható út, hogy a munkát a géppel végeztetjük

ugyan, de numerikusan bizonyítjuk, illetőleg cáfoljuk az egyenlőtlenséget.

### Numerikus bizonyítás

Az egyszerűség kedvéért zárt szakaszon értelmezett, egyváltozós, differenciálható függvényekkel foglalkozunk. El kell döntenünk, hogy igaz-e ezek között például a „kisebb" vagy a „kisebb vagy egyenlő" reláció. Ismerjük a gép által kiszámított helyettesítési értékek pontosságát, az első differenciálhányadosok abszolút értékének felső korlátját — mindkét függvény esetében. A bizonyítás, helyesebben a bizonyítási kísérlet a következőképpen zajlik.

Mindkét függvény helyettesítési értékét kiszámítjuk ugyanabban a tetszőleges pontban. Ezek hibáit ismerve, meg tudjuk állapítani, hogy a szóban forgó pontban igaz az egyenlőtlenség, nem igaz az egyenlőtlenség, vagy nem dönthető el a kérdés. A harmadik eset a kritikus, mert előfordulhat, hogy több kísérlet után sem akadunk olyan pontra, ahol eldönthető lenne az egyenlőtlenség. Ebben az esetben ez a módszer nem használható.

Ha van egy olyan pontunk, ahol az egyenlőtlenséget el tudjuk dönteni, és nem igaz az egyenlőtlenség — akkor munkánk befejeződött.

Ha viszont olyan pontot találtunk, ahol a kérdés eldönthető, és igaz az egyenlőtlenség, akkor folytatnunk kell a munkát. Legyen ez a pont x. A függvények differenciálhányadosa abszolút értékének felső korlátait ismerve kijelölhető egy x középű szakasz, amelyben biztosan igaz az egyenlőtlenség.

E szakasz végpontjaiban újra megvizsgáljuk az egyenlőtlenséget, és az iménti elvek szerint vagy eredménytelenül kell abbahagynunk a munkát, vagy megcáfoltuk az egyenlőtlenséget, vagy pedig újra módunk nyílik bővíteni azt az intervallumot, amelyen fennáll az egyenlőtlenség.

Ezekkel a bővítésekkel vagy ki lehet meríteni az eredeti intervallumot véges (és nem túl sok) lépésben, vagy nem. Ha igen, akkor „szerencsésen" bebizonyítottunk egy egyenlőtlenséget, esetleg egy olyan egyenlőtlenséget, amelynek klasszikus módszerekkel való bizonyítása reménytelen lenne.

### Az elefántcsonttorony és a „tökéletlenség"

Egy sokak által megoldani kísérelt problémát — töméntelen számolás és vizsgálat elvégzésével — végre megoldottak! Méghozzá géppel. Megdöbbentő volt az eredmény fogadtatása. Nem a kitörő öröm, hanem a fanyalgás volt a jellemző. A megoldást ugyanis nem a „brutális erőtől" várta a szakma, hanem valahogyan másképp. Szellemesen, szépen, virtuózan, mélyenszántóan. Ha így nem megy, jobb lett volna (tán), ha marad a sejtelmes bizonytalanság, a fájó tudatlanság...(?)

A számítástechnika a tudomány rendkívül hatékony új eszköze. Kétségtelen, hogy ez a hasznosság nagyrészt megbízható robotolásban valósul meg. Kevés az esztétikum. Nem biztos azonban, hogy ennek a számítástechnika az oka. Lehet, hogy még nem foglalkoztunk elég alaposan a hibákkal és a megbízhatósággal. Lehet, hogy ha majd ezeken a területeken is többet tudunk, akkor másképp fogjuk látni még a tudományon belül is az esztétikumot.

**Pogány Csaba**

---

## Sokváltozós függvény lokális minimuma

# A poli(p)technikai oktatás

A nyílt vízen úszó polip egyik karja zsákmányt érez...
Az imént még szerteszét nyúló többi kar is lecsap: a polip körülfogja az áldozatot...
A ragadozó összegömbölyödve viszi el a mélyebb régiókba, a búvóhelyére.
A háromdimenziós poliéder (triéder) olyan, mint egy háromlábú fényképezőgép-állvány (egy háromkarú polip). Az állvány lábait kihúzva-betolva akármilyen gödrös talajon fel tudjuk állítani. Valamelyik lába a szűkebb környék legmélyebb pontjához kerül.
Ugyanígy, a sokdimenziós poliéder ügyes módosítgatásával megtalálhatjuk „a sokdimenziós felület mélypontját".

Műszaki-tudományos számítások során igen gyakran kell megkeresni valamilyen függvény helyi minimumát. (A maximumkeresés nem más feladat, hiszen f(x) maximuma ott van, ahol -f(x) minimuma.) Az iskolában tanítják, hogy minimum ott van, ahol

$$\frac{df(x)}{dx} = 0; \qquad \frac{d^2f(x)}{dx^2} > 0,$$

tehát a minimum helye — legalábbis elvileg — könnyen megtalálható. Többváltozós esetben ugyanígy lehetne eljárni, de a feltétel második részének teljesítését többnyire nem egyszerű vizsgálni: a skalár helyett mátrix az elemzés tárgya, ami egyfelől sok deriváltfüggvény-érték kiszámítását igényli, másfelől pedig meg kell határozni a mátrix sajátértékeit is. A közkézen forgó programok emiatt általában a gradiens-módszert használják: az első deriváltakat tartalmazó vektor a leggyorsabb függvényérték-növekedés irányát tűzi ki, vele szemben haladva előbb-utóbb eljutunk ahhoz a ponthoz, amit minimumnak vélünk. (Lehet, hogy csak inflexió!) Mégsem mondhatjuk, hogy ez a módszer megoldja feladatainkat, hiszen nem minden függvénynek van deriváltja. Gyakori ez a szerkezet:
if feltétel then f(x):=g(x) else f(x):=h(x)
de az ilyen függvénynek nem tudjuk a gradiensét programozni. Szükség van

tehát olyan minimumkereső eljárásra is, amelyik csak a függvény értékeit használja fel.

## Minimumkeresés flexibilis poliéder módszerrel

A módszer alapgondolatát J. A. Nelder és R. Mead 1964-ben publikálta. A magyar számítástechnika hőskorában, a legendás RAZDAN-3 gép ALGOL-60 könyvtárában Ugray L. 1974-ben írt gépi kódos programja volt elérhető. Az algoritmusnak ma is friss szellemessége és egyszerűsége indokolja közkinccsé tételét.

A bemutatást egy Fortran77 nyelven írt szubrutin szolgálja. Ennek a nyelvnek a választását az indokolta, hogy (például Pethő Ádám C-ről írt könyvében kifejtett véleményével ellentétben) ez az egyetlen olyan, széleskörűen elterjedt nyelv, amelyik lehetővé teszi a programok transzportját: az American National Standard Institute által kiadott ANSI X3.9-1978 szabvány definiálja szabályait. Ha a programban nem alkalmazzuk a fordítóprogramok készítői által kínált kényelmesebb megoldásokat, a program bármelyik, Fortran77-et ismerő gépen fut. A szubrutin a polyp.for, a mintafeladat főprogramjának legfontosabb része a POLIP.for, a futtatható mintaprogram a POLIP.EXE, a CGA/EGA/VGA monitoros gépeken futtatható rajzos működésbemutató pedig a POLIR.EXE fájlban található. (Ezek — a módszer részletes leírásával együtt — megvásárolhatók az Alaplap Lemezek sorozatban.)

## Mikor poliéderekkel, mikor máshogyan?

A Polyp eljárás közepesen sok változós függvényekre kidolgozott algoritmust valósít meg (n nem több, mint 10...15), de működőképes más esetekben is. Más kérdés, hogy érdemes-e használni.

Ha n igen nagy, rendkívül sokfelé haladva kereshetjük a függvény minimumhelyét. Nagyon távoli kezdőpontból indulva, vagy a kezdeti poliéder élhosszát balszerencsésen felvéve a kerekítési hibák miatt az igazi minimumtól távol is abbamaradhat az iteráció, vagy pedig nem történik érdemi keresés, csak JT nő MXIT eléréséig. Célszerű ilyenkor a kezdeti értékeket hatékonyabb módszerrel előállítani, ami sokszor eseti programozást jelent. Máskor az utolsó lehetőséget kell választanunk: sokváltozós függvények minimumhelyének Monte-Carlo-keresését. (Véletlenszerűen veszünk fel pontokat a sokváltozós térben. A program „megtanulja", merre kell haladnia, hogy a minimumot megtalálja.) A Monte-Carlo módszer 50-60 változó felett már biztosan gazdaságosabb, mint a Polyp.

Ha a változók száma kicsi (mondjuk 5 alatti), egyszerű függvények esetén a gradiensmódszer gazdaságosabb a Polypnál. Ha csak egyetlen IF is van a függvényben, akkor viszont már a Polypot kell választani. (A Polyp működését bemutató mintafeladat is egy olyan függvény minimumhelyének keresése, amelyiknek a deklarációja IF-et tartalmaz.)

## Az algoritmus

Az algoritmus működését kétdimenziós példával lehet legkönnyebben bemutatni. (Az algoritmus szabatos leírása a polyp.for 15.-44. soraiban található. A polir.exe moziszerűen generálja az ábrákat.) Az 1. ábra ellipszisei fürdőkád alakú völgy térképének szintvonalai. A tetszőlegesen felvett kezdőpontból a két koordinátatengellyel párhuzamosan kinyújtjuk a poliéder kezdeti oldalait. Kiszámoljuk a három pontban a függvény értékét, majd úgy számozzuk be a pontokat, hogy f f f (!!!) legyen. A 2-es és 3-as pontból 1 2 3 álló részrendszer súlypontjának kiszámoljuk a koordinátáit, majd erre tükrözzük az 1-es pontot. Ha itt kisebb függvényértéket találunk, mint az eddigi minimum, még egy lépéssel tovább is megyünk, de az 1-es pontot mindenképpen eldobjuk. A 2. ábra ezt az esetet mutatja be. A pillanatnyi poliéder végpontjaiban kapott függvényértékekkel ugyanúgy járunk el, mint az előző lépésben tettük. Abban az esetben, ha a tükrözéssel túlszaladnánk a minimumon, az új pontot a súlypont és a tükrözött pont között vesszük fel. Ha a függvényérték még itt is túl nagy, végső megoldásként a minimumpont köré zsugorítjuk a poliédert, ahogy a 3. ábrán látható.

## A mintafeladat

Az ún. „HPGe spektrométer" mint berendezés kalibrációjánál fontos lépés annak ellenőrzése, hogy (az itt nem részletezett fizikai folyamatok miatt) hogyan lesz az egyetlen függőleges vonalból a 4. ábrán bemutatott görbe. A görbe egy jobbra lejtő egyenesnek és egy, két részletben definiált függvénynek összege. A jobb oldal Gauss-haranggörbe, a bal pedig exponenciális függvény. Ezt a függvényt 6 paraméter írja le (argumentum a fotonenergia): a csúcs magassága, a csúcs helye, a haranggörbe inflexiójánál mért fél szélesség, a haranggörbe és az exp-görbe átmenetének (a töréspontnak) a helye, a (háttér)egyenes elsőfokú együtthatója, a (háttér)egyenes konstans tagja.

A polip.exe program a felhasználó által adott adatok alapján keresi meg a fenti hat paraméternek azt az együttesét, amelyik az elérhető legjobb illeszkedést biztosítja. Azoknak a kedvéért, akik nem rendelkeznek a megfelelő fordítóprogrammal, az átkódolást megkönnyítendő a program néhány megoldását más nyelveken is bemutatjuk. (Az Alaplap Lemezek említett lemezén kívül e számunk mellékletén a „Poli(p)technika matematikából" c. alatti anyagok.)

**Szondi Egon János**



*– Főorvos úr, a szívverés ugyan leállt, de még van 2 életünk...*

Jó döntés és következmény: virágzó farm

# A gazda szeme

Az átalakuló mezőgazdaság új kihívásokat támaszt az agrár-ökonómiai kutatásokkal szemben.
A nagyüzemi struktúra felbontásával párhuzamosan mind nagyobb hangsúlyt kapnak a farmok vezetését segítő menedzsmenteszközök.
A cikkben arról a gazdaszemről szeretnénk szólni, amelyik gondosságával, odafigyelésével „hizlalja a jószágot".
Ez a tekintet Magyarországon is egyre inkább keres-kutat olyan, az alkalmazott tudományos kutatás eredményeit magában foglaló szoftvereket is, amelyeket a gazdasággal való törődésben megbízható partnerként tudhat maga mellett.

A mezőgazdasági vállalkozások, farmgazdaságok irányításában, menedzselésében alkalmazható szoftverek még a viszonylag szerves fejlődést felmutató Nyugat-Európában is csak hosszú folyamat eredményeként váltak a mindennapi gyakorlat részévé. Ez a folyamat hazánkban a '70-es évek elején vette kezdetét, azonban a mostani átalakulás minden bizonnyal új, a nyugat-európaihoz hasonló irányba tereli a menedzsmentszoftverek alkalmazását.

## Döntéstípusok a „földön"

Ebben a rövid áttekintésben először a farmmenedzsmentnek azokat a döntési típusait gyűjtjük csokorba, amelyek joggal várhatnak el valamilyen szoftvertámogatást, majd sorra vesszük azokat a tipikus szoftvereket, amelyek rendszerint minden nyugati katalógusban megtalálhatók, s röviden jellemezzük őket a hazai felhasználhatóság szempontjából.

— A gazdálkodás elindítása, felfuttatása, szűkítése, megszüntetése. (1.)

— Mit termeljek? (2.)

— Technológiaválasztás: elsősorban kézierőre vagy inkább gépekre alapozott termelés? (3.)

— A hozamok megtervezése: milyen hozamszinten várható a legnagyobb profit? (4.)

— Milyen szaporítóanyagot vásároljak, hogyan alakuljon általában az anyag- és energiafelhasználásom? (5.)

— Az értékesítési csatornák kiválasztása, szerződéskötés. (6.)

— A piaci áringadozásokkal és a kölcsöntőke kamatainak változásával összefüggő kockázat kezelése. (7.)

— Biztosítás. (8.)

— A vállalkozás pénzügyeinek menedzselése. (9.)

— Hosszú lejáratú hitel(ek) megnyerése. (10.)

— Beruházási döntések. (11.)

A felsorolt ökonómiai döntéstípusok döntéshozatali folyamata kisebb-nagyobb mértékben standardizálható. A standardizáltság mértéke — ami egyúttal az algoritmizáltság lehetőségét is jelenti — nem feltétlenül elsődleges szempont a folyamat számítógépi támogatottságában. A mezőgazdálkodás esetében különösen hangsúlyozott szerepe van az egyéni intuíciónak, hiszen ez a tevékenység élő szervezetek (növények és állatok) növekedésén alapszik, valamint a természeti környezet (elsősorban az időjárás) közvetlenül befolyásolja. Mindezek eredőjeként a véletlen lényegesen nagyobb hatást fejt ki a gazdálkodásra, mint egyéb tevékenységek esetében, s így a döntéshozó beleélő képessége, megérzései és ösztönei fokozottabb szerephez jutnak. A számítógép elsődlegesen nem a folyamatok irányításában, hanem sokkal inkább a folyamatok egyre komplexebb megfigyelésében és paramétereinek rögzítésében, illetve az adatok, információk megfelelő rendszerezésében lehet a gazda partnere. Nézzük röviden, milyen jellemző szoftverkínálatot találhat az érdeklődő ezen a téren.

## Farmmenedzsment — szoftverekkel

A közelmúltban alkalmunk nyílt néhány szoftver tesztelésére egy megbízás alapján. (A tesztelés részleteire nem térünk ki, mert arról tartjuk fontosnak a tájékoztatást, hogy ami van, mire használható — hogy a közismerten szűkös kutatási/fejlesztési erőforrásaink ne aprózódjanak el már lefedett területekre.) Az 1991-es német agrárközgazdasági szoftverkatalógust áttanulmányozva azt bocsátjuk előre, hogy ez a kiválasztás (a szoftverek elterjedtsége alapján is) eléggé jellemzőnek tekinthető. A felsorolás ezért most nem is annyira az egyes programok, mint inkább a megfelelő szofvertípusok összevetését szolgálja.(A sorszám az előző felsorolásra utal.)

A táblázatból az a fontos tanulság szűrhető le, hogy a meglévő szoftverek rendkívül erős hangsúlyt helyeznek a gazdálkodás termelési oldalára. Úgy tűnik, hogy a termelés szakmai fogásai azok, amelyek a jó gazda gondosságaként elsősorban megnyilvánulnak. Ezen szemlélet erősítésére nálunk is égetően szükség van, hogy valamelyest helyreálljon a gazda és a föld—növény—állat korábbi harmonikus kapcsolata.

## Van (jó) magyar termék is!

Ugyanakkor látnunk kell azt is, hogy a nyugati országokban pénzügyi és adótanácsadó cégek egész sora kínálja szolgáltatásait a farmereknek, s így a döntéshozatalt nem kell feltétlenül minden részletre kiterjedően szoftverrel támogatni. Nálunk a hasonló tanácsadás egyrészt még nem célozta meg a farmerokat, másrészt a farmgazdálkodás jövedelemtermelő potenciálja nem is bírja még el az egyéni tanácsadás költségeit. Inkább szaktanácsadó hálózat kiépítésével lehetne a folyamatot fellendíteni.

Másik fontos tapasztalatunk az volt, hogy a szoftverházak a gazdálkodás egyre több területét lefedő komplex

| A szoftver neve | Rendeltetése | Érintett probléma |
|---|---|---|
| HKS Power Fibu | Könyvvezetés, pénzügy | 1, 2, 7, 9, 10, 11 |
| HKS Lohnunternehmer | Szállító-/vevőnyilvántartás, számlázás | 6, 9 |
| HKS Schlagkartei | Táblatörzskönyv | 1, 2, 3, 4, 5 |
| HKS LandRat | Földbérlet/vásárlás | Nem standardizált |
| HKS Elfriede | Szarvasmarhatartás | 1, 3, 4, 5, 9 |
| HKS Pig Check | Sertéstartás | 1, 3, 4, 5, 9 |
| HKS Düngeplan | Szerves- és műtrágyázás | 4, 5 |
| Siemens Ackerschlag | Táblatörzskönyv | 1, 2, 3, 4, 5 |
| Siemens Düngeplanung | Szerves- és műtrágyázás | 4, 5 |
| Siemens Einnahmen Ausgaberechnung | Pénzügyi nyilvántartás | 9 |
| Siemens Finanzbuchhaltung | Könyvvezetés, pénzügy | 1, 2, 7, 9, 10, 11 |
| Siemens Sauenplaner | Sertéstartás | 1, 3, 4, 5, 9 |
| Siemens Rinderplan | Szarvasmarhatartás | 1, 3, 4, 5, 9 |
| Land Data Schlagkartei | Táblatörzskönyv | 2, 3, 4 |
| GATE Flóra | Növénytermesztés | 1, 2, 3, 4, 9 |
| GATE Bovis, Granum, Mixer | Tejtermelés | 1, 2, 3, 4, 9 |
| GATE Pig-Control | Sertéstartás | 1, 2, 3, 4, 9 |
| GATE Invest | Beruházás | 8, 9, 10 |

csomagokat kínálnak. Ezt figyelhettük meg mind a Siemens, mind a HKS (Horn-Köhler Software and Computer Systeme GmbH) vonatkozásában. A Gödöllői Agrártudományi Egyetem (GATE) hasonlóan ebbe az irányba tett lépéseket. Az integráltság tekintetében más a helyzet: egyrészt a teljes integráltságot egyik gyártó sem valósította meg, másrészt általában nem is fogalmazzák meg kinyilvánított célként. Ez a törekvés leginkább a termelés szakosodására vezethető vissza, aminek a szoftver oldaláról a független, önálló modulok felelnek meg főként.

A vizsgált szoftvereknek közös vonásuk volt a felhasználóbarát menükezeléses technika, az online help lehetősége, valamint a rugalmas alkalmazkodás képessége a helyi sajátosságokhoz. Mindegyik program minimális hardverkiépítettség (256 kbájt, két meghajtó és valamilyen grafikus kártya) esetén már installálható és működőképes.

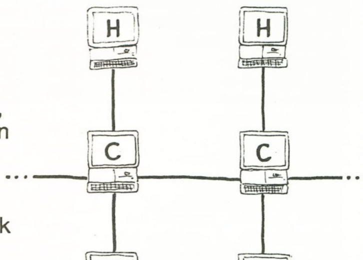
Végezetül szeretnénk abban bízni, hogy a címben aposztrofált gazda látása a felsoroltakhoz hasonló eszközök alkalmazásával egyre élesedik, gondoskodása ennek következtében is javul, s a gyarapodó jószágok révén nemcsak a farmer, hanem valamennyien ennek hasznát látjuk.

**Ferenczy Tibor—Tóth József**

## Szerves kapcsolat a „szervetlen” számítógéppel

# Gyógyszermodellezés

Egy svájci gyógyszergyár vezető menedzsere panaszkodik, hogy a biológusok és gyógyszervegyészek az utóbbi időben idejük egyre nagyobb részét nem a laboratóriumban, a munkapad és a műszerek mellett, hanem a számítógépek képernyői előtt töltik. Hogy lehet ez? — csodálkozhatnánk — hiszen a biológusok élő szervezetekkel és az élő anyag tulajdonságaival foglalkoznak, a vegyészek pedig vegyületeket állítanak elő lombikokban...



P(V)C

Arra gondolnánk, hogy az „élet-tudósok” és a „méregkeverők” valószínűleg csakis azért ülnek számítógépnél, mert egy közleményt vagy jelentést írnak szövegszerkesztővel, előadási ábráikat rajzolják egy grafikus programmal, online kapcsolatban vannak egy adatbankkal, mert irodalomkutatásba merültek el, éppen az adataikat viszik be egy adatbázisba, vagy rendezik egy táblázatkezelővel, eredményeiket értékelik egy statisztikai programcsomag segítségével, szakértő rendszert használnak, vagy a számítógép-hálózaton keresztül egy másik kutatóval kommunikálnak — hogy csak a tudományos kutatásban előforduló általános számítógép-alkalmazásokat említsem. Sokszor azonban e triviális tevékenységeket kísérő jelenségek helyett bonyolult, színes ábrákat látunk nagy felbontású grafikus képernyőn, a számítógépek pedig nagy teljesítményű RISC processzoros gépek, VAX munkaállomások, vagy pedig éppen távolabb elhelyezett Cray vagy más szuperkomputerek.

Egy viszonylag új fejlemény, az ún. 3D-QSAR (háromdimenziós, kvantitatív szerkezet—hatás összefüggések) módszereinek alkalmazása az, aminek szemtanúi lehetünk.

Ezeket a számítógépes technikákat enzimek és más biopolimerek, például a genetikai információt hordozó DNS-ek és kisebb molekulasúlyú szerves molekulák modellezésére, vizualizálására, valamint tulajdonságaik és kölcsönhatásaik vizsgálatára alkalmazzák a biológiában, kémiában, valamint a gyógyszer- és növényvédőszer-hatóanyagok kutatásában. A kémiai szerkezet és biológiai aktivitás kvantitatív

összefüggéseinek kutatását először a számítógépek elterjedése tette lehetővé, és ez az irányzat mintegy 25 éves múltra tekinthet vissza.

Ezeket az eljárásokat az angol „Quantitative Structure Activity Relationships" elnevezés rövidítéseként QSAR módszereknek nevezik. Ebben az irányzatban — amelyben molekulamodellezést még nem alkalmaztak — a gyógyszermolekula szerkezetének a biológiai hatásban fontos tulajdonságait és szerkezeti vonásait számszerűsítik (kémiai deszkriptorok) és hozzák összefüggésbe a biológiai hatás mért, számszerű értékeivel.

A leggyakrabban alkalmazott matematikai mószerek a lineáris, bilineáris és parabolikus regresszióanalízis, valamint egyéb korrelációs technikák és illesztések, az ún. Free-Wilson-analízis, többváltozós matematikai módszerek — mint például diszkriminanciaanalízis, clusteranalízis, faktor-, fkomponens- és kanonikuskorreláció-analízis, részleges legkisebb négyzetek módszere rejtett struktúrák analízisére (partial least squares for latent structures), az adaptív legkisebb négyzetek módszere (adaptive least squares) —, valamint alakzatfelismerési eljárások — mint például az „N-legközelebbi szomszéd" (N-nearest neighbour) és a „nemlineáris tanulógép" (nonlinear learning machine) algoritmusok stb.

Újabb fejlemény az „ideghálózatok" (Neural Networks) algoritmusainak alkalmazása nemlineáris összefüggések kezelésére.

Az összefüggéseket leíró matematikai modell segítségével tanulmányozni lehet a gyógyszereknek a biológiai hatás kifejtéséhez szükséges szerkezeti tulajdonságait, hatásmechanizmusát, és prediktálni lehet a tervezett, még elő sem állított analóg szerkezetű molekulák biológiai hatását. A QSAR-módszereknek elsősorban a gyógyszermolekula szerkezetének szisztematikus módosításával történő hatásoptimalizálásban van jelentőségük. A QSAR-elemzések alkalmazásával elérhető megtakarítás nagyságrendjét akkor mérhetjük fel, ha tudjuk, hogy mihez viszonyítsunk: e módszerek nélkül ma mintegy harmincezer különböző, erre a célra előállított vegyületből lesz egyetlenegy kereskedelmi gyógyszer; a gyógyszerkészítmény kifejlesztése átlagosan mintegy 120 millió dollárba kerül (ebben benne van a 30 000-féle előállított molekula ára), és 8-10 évig tart. A QSAR módszerek alkalmazásával a létrehozandó vegyületek száma mintegy felére, harmadára csökkenhet.

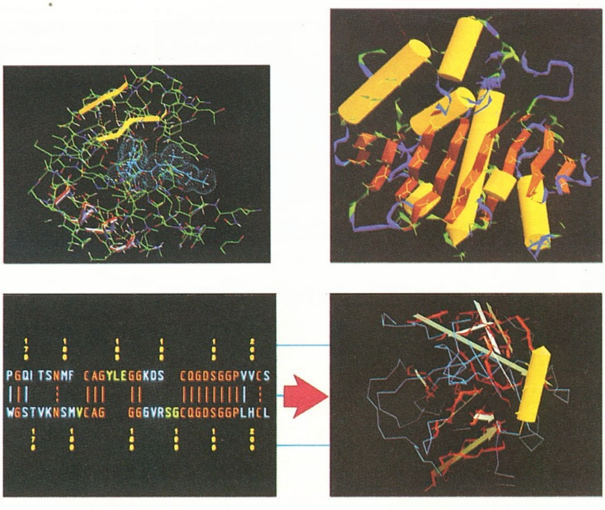
A klasszikus QSAR-alkalmazásoknak, noha ma is elterjedten használják őket, az a hátrányuk, hogy a biomakromolekulák tulajdonságait, az egymással és a gyógyszermolekulákkal való kölcsönhatásait és az ekkor kialakuló, bonyolult térbeli viszonyokat nem tudják adekvátan leírni és vizualizálni. A gyógyszermolekulák ugyanis legtöbbször enzimekkel, nagy molekulasúlyú, bonyolult térbeli struktúrájú aminosavpolimerekkel, ún. fehérjékkel vagy más biopolimerekkel lépnek kapcsolatba, azokat gátolják, vagy éppen serkentik. A növényvédőszerek gyakran egy kártevő életfontosságú enzimjét bénítják azzal, hogy a hatóanyag-molekula szorosan a receptorhelyhez kapcsolódva gátolja azt.

Óriási fellendülést hozott a klasszikus, numerikus QSAR programok mellett a biopolimerek modellezésére és vizualizálására alkalmas 3D-QSAR módszerek megjelenése. E molekulamodellező és molekulárgrafikai programok már 1992-ben egymilliárd dolláros piacot jelentenek, ami 1996-ra várhatóan meg fog duplázódni.

A szimulálandó molekulák és kölcsönhatásaik bonyolultsága miatt azonban a 3D-QSAR módszerek rendkívül költséges hardvert és szoftvert igényelnek. Azt lehet mondani, hogy a számítógép nem lehet eléggé gyors: egyre nő a biológiában a teraflop ($10^{12}$ lebegőpontos művelet/s) teljesítményű gépek iránti igény (ma egy ilyen számítógép építése már nem lehetetlen, és költségét mintegy 100 millió dollárra becsülik). A 3D-QSAR programok ára néhány tízezertől több százezer dollárig terjed. A 3D-QSAR kutatások zömét azonban közepes teljesítményű VAX munkaállomásokon és RISC-processzoros miniszámítógépeken (például Silicon Graphics Personal IRIS) végzik.

Belátható, hogy a bonyolult biomolekulák grafikus megjelenítése miért rendkívül számítás- és hardverigényes, hiszen a vizualizálás — a számítástechnika egyik állandóan alkalmazott megjelenítési módja — nagy erőforrás-fogyasztó. De hogy miért olyan számításigényes a biopolimerek és kölcsönhatásaik modellezése? Hát..., erre bizonyító erejű válasz egy párhuzam legyen (lásd a keretes részben).

Mindazonáltal a nem túl nagy adatmátrixok esetén a fent felsorolt numerikus eljárásokat alkalmazó QSAR programok (például Drugidea 2.01) már XT és AT kategóriájú gépeken is futtathatók. A molekulárgrafikai programok által elvárt magas hardverigényt a személyi számítógépek természetesen alulról közelítik. Vannak olyan egyszerű molekulamodellező programok, amelyek már AT 286-os gépeken is megélnek, de elfogadható sebességgel csak 386/486-os gépeken futnak (486-os gé-




*1. ábra. A Chem-X program ChemProtein modulja által alkotott modell, amely egy fehérje és egy gyógyszermolekula kölcsönhatását vizualizálja. A fehérjének és a gyógyszermolekulának csak a hidrogénatomoktól megfosztott gerince van feltüntetve, a gyógyszermolekula ún. Van der Waals-felszíne pedig kék raszterpontokkal van jelölve.*

## Az élet és az abszolút sakkjátszma

Jó néhány évvel ezelőtt olvashattunk egy cikket a New Scientist című angol folyóiratban „Intrinsically unsolvable problems", azaz „Lényegükből következően megoldhatatlan problémák" címmel, amelynek alapján bárki számára érzékletessé válhattak a számítógépeknek a — hosszú távon beláthatatlan fejlődési lehetőségeik ellenére megmaradó — korlátai. E cikk az „abszolút sakkjátszma" számítógépi lejátszásának lehetőségét tárgyalta. Azaz azt, hogy végigjárható-e reális időn belül az a fa, amely tartalmazza egy sakkjátszma összes lehetséges első lépését (20 lépés), minden egyes első lépésre adott összes lehetséges válaszlépést (400 lépés) és így tovább, amíg a játszma befejeződik. Könnyen kimutatható, hogy az elemezendő lépések száma hamarosan óriásira nő, és bekövetkezik egy ún. kombinatorikai robbanás.

A fa bejárásának számításigényességét a szerző a következőképpen érzékelteti: ha egy akkora méretű számítógépet alkalmaznánk, mint a jelenlegi elképzelés szerinti világegyetem, s amelyben egy félvezető akkora lenne, mint egy proton, a gép kapcsolási sebessége olyan lenne, mint a jelenlegi leggyorsabb kapué stb., stb., nos, ebben az esetben is a nap már régen kihűlne, mire a számítógép a teljes fát kiértékelné.

Tehát soha nem lesz akkora teljesítményű számítógépünk, amely ezt az alapjában véve egyszerű — 64 kockán, 32 figurával és véges számú szabállyal leírható — játékot/feladatot kimerítően megoldaná. Márpedig a sakkjátékhoz — akár mint kedvelt mesterségesintelligencia-problémához — képest egy közönséges vízmolekula lehetséges állapotainak kvantumkémiai leírása is sokszorta bonyolultabb feladat. Ha pedig az élő anyag hatalmas molekuláit és máig is csak részben értett bonyolult struktúráját tekintjük, akkor bele kell törődnünk, illetőleg fellélegezhetünk (nézőpont kérdése): bármekkorára is nő idővel a számítógépek teljesítménye, a modellalkotás és a heurisztikus emberi godolkodás mindig is a megismerés legfontosabb eleme marad. A számítógép ugyanakkor egyre hatékonyabb és nélkülözhetetlenebb eszköz lesz az ember által kigondolt absztrakt modellek, például molekulamodellek megalkotására és alkalmazására a kutatásban.

pen a numerikus sebesség 30-40-szer, a grafika sebessége mintegy 8-szor gyorsabb, mint egy 286-os gépen). Ilyenek például a Molidea 2.2 vagy a Moby 1.4. és egy sor más molekulamodellező program. A Molidea és a Drugidea fejlesztése Magyarországon a CompuDrug kisszövetkezetben kezdődött, majd a programozócsoport kiválása után a CheMikro Kft.-ben folytatódott. A hardverkiépítés alsó határa, ahol már molekulák kölcsönhatásait is elemző modellezőprogramok használhatók: AT-386, 33 MHz, SVGA színes monitor, aritmetikai koprocesszor. Minél nagyobb RAM és disk-caching a merevlemez-műveletek gyorsítására sokszor előnyös.

Jó színvonalú, érdemi munkára használható például a Chem-X moduláris programcsomag (Chemical Design Ltd., Oxford), amely futtatható az IBM PC 386/486 kompatibilis és az Apple Mac II gépeken. A Chem-X-et eredetileg VAX gépekre fejlesztették ki (VMS, majd később Unix operációs rendszer alatt) VaxStation, Silicon Graphics, IBM PC és Apple Mac II grafika alkalmazásával). (Lásd az *1. ábrát.*) A Chem-X ára akadémiai intézetek számára modulonként £250, és a minimális kiépítés, amivel már dolgozni lehet: négy modul.

Olcsóbb, mintegy 400 dollár az Allinger PC Model molekulamodellező programja, amely egyszerűbb, rudimentális kivitelezése miatt elsősorban a szakterülettel ismerkedők számára ajánlható.

Nem e szakterületi PC-alkalmazók körében ismert az Alchemy II (Tripos Associates), amely egyszerűbb molekulamodellezési feladatok elvégzésére használható. Az egyik legújabb és legkorszerűbb molekulamodellező program a Hypercube cég által épített magas színvonalú, integrált molekulaszerkesztő program, a HyperChem (Autodesk), Windows környezetben, standard AT-386/486 számítógépekre.

Az egyszerűbb molekulamodellező programok (például Molidea) kiválóak elemi szinten a kémia oktatása céljára, mert a szerves molekulák térbeli tulajdonságait és bizonyos más tulajdonságait — például: izoméria jelensége — jól szemléltetik. (Izomerek: azonos számú és azonos atomokból felépülő, de az elrendezésüktől függően egymástól különböző térbeli szerkezetű vegyületek — például: D-tejsav és L-tejsav.)

Kifejezetten oktatási célra kifejlesztett program azonban a Desktop Molecular Modeller (DTMM), amelyet S. Aduldecha és munkatársai ismertetnek egy kémia oktatásával foglalkozó folyóiratban (J. Chem. Educ. 1991, 68(7), 576-583). Lehet tőlük különlenyomatot kérni. (Nem tudok arról, hogy ilyen oktatási program kereskedelmi forgalomban lenne.)

**Bordás Barna**

ANT LTD.



*SyQuest cserélhető lemezes winchesterek 44, 88 Mbájt*

**ANT Ltd.**
Budapest, József krt. 70. I/5.
1399 Bp., Pf. 701/349
Tel./fax: 133-1670

Szerettel várjuk látogatóinkat a COMPFAIR-en, az A pavilon 203-as standján.

Az ANT Ltd. a SyQuest Technology kelet-európai disztribútora.

INFORMÁCIÓKÉRÉS: 66 ▼

# SHARP

3% kedvezményre jogosít vásárlás vagy szolgáltatás igénybevételekor

## Másológép centrum

***Példa árainkból:***

SHARP Z 30 39 900,– + áfa
SF 6100 74 900,– + áfa
SF 7370 113 900,– + áfa
SF 7850 173 900,– + áfa
Minden géphez indulókészlet: 15.900,– + áfa

- Másológépek forgalmazása és szakszervize.
- Kellék-, festék-, papírellátás.

Canon, Mita, Ricoh, Toshiba másológépekhez festékek és kellékek DISZKONT ÁRON.




E·Copy Kft.
Fénymásolók értékesítése és szakszervize

1146 Bp. XIV.,Thököly út. 57/B
Tel.: 251-1869, 252-2566

INFORMÁCIÓKÉRÉS: 67 ▼




...gyors emelkedés

# PERON · REKLÁM Kft.

Telefon: 149-4819

INFORMÁCIÓKÉRÉS: 68 ▼

FAST




Authorized Distributor

**Multimédia rendszerek**
**E I Z O monitorok**
**H P perifériák**

VIDEO MACHINE

Desktop Video Studio
az Ön PC-jén is !
Premier :
Compfair A/204
92. Okt. 6-10.

ALLEGRO
1016. Budapest, Tigris u. 28.
Tel.,Fax : 15 68 132, 17 55 404

INFORMÁCIÓKÉRÉS: 03 ▼

## Orvosinformatikai rendszerek

# A medicina „szívében”

Az orvoslás legszebb feladata a betegségek megelőzése lenne, hiszen a gyógyítás már csak a bekövetkezett baj káros következményeinek mérséklésére törekedhet. A lakosságszűrés, a kórmegelőzés azonban csak részben az orvos (és még kisebb részben a számítástechnika) felelőssége, mert ahhoz az egész társadalom, közösség együttműködése szükséges. Ennek elnyerése viszont inkább lélektani, neveléstani és reklámtevékenységek eredménye lehet. Mindazonáltal a már veszélyeztetett állapot és a betegség ellátásában, kiújulásának megakadályozásában — éppen az adatok tömege és bonyolultsága miatt — a számítógéppel támogatott orvosinformatika óriási segítséget nyújt.

A medicina tudomány és szolgáltatás, de egyiknek sem szokványos. Mint tudomány legnagyobb részt empirikus, kvalitatív, természetrajzi, leíró, és csak kisebb részben kvantitatív, természettudományosan szabályfelismerő és -alkalmazó. Mint szolgáltatás csupán valószínűsíti, de egyáltalán nem garantálja az eredményt, még ha ezt nem is szívesen veszik számításba a rászorulók. Ebben mentsége éppen a már említett empirikus alapja, és hogy „hozott anyagból dolgozik”; ám ez nem passzívan tűri, hanem aktívan segíti vagy éppen gátolja a munkát. Ráadásul nem a mester választja meg a feladatot, hanem a „feladat” — éppenséggel hozzáértés nélkül — választ „megoldó”-t, jól-rosszul. Mindehhez még nehezítő körülményként járul, hogy a munkaeszközök egy mestertől, műtől független harmadiknak, az ellátónak az adományai, méghozzá szerteszét telepítve.

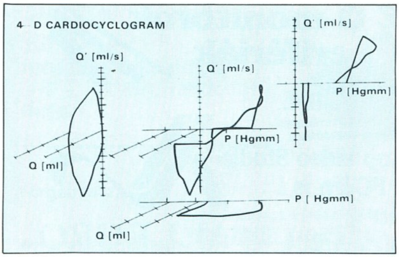
Ilyen körülmények között az eredmény valószínűsége a tudás mint alap, az alkalmazás/szervezés és az eredményfelhasználás optimalizációjának sikerétől függ. Ennek a rendkívül bonyolult munkának egyik kezdeti, de fontos része a tünetből diagnózis folyamat közelítése valószínűségszámítási módszerekkel, az optimalizáció komplett csúcspontja pedig: szimulációs modellek működtetése.

Az orvosi munka folyamatának sokfélesége, az egyes diagnosztikus vagy terápiás lépések, esetleg lépéskombinációk kockázatának ismerete (információ) és bekövetkeztének esélye (valószínűség) nyilvánvalóvá teszi, hogy e hivatás gyakorlásában az információelmélet és a valószínűségszámítás között fontos kapcsolat van. Csak egy egyszerű példa: ha S tünet (például mellkasi fájdalom) esetén a D1 diagnózis (koszorúér-meszesedés) fennállásának valószínűsége kétharmados, azaz P[D1:S]=0,66 és ugyanakkor D2 (=nem koszorúér-betegség) valószínűsége egyharmad, azaz P [D2:S]=0,33, akkor ha véletlenül kiválasztott S tünetű, 120 páciens mindegyikére kimondom a D1 diagnózist, 80-szor igazam lesz, ami nem rossz arány egy tömegvizsgálati szűréskor. De 40 esetben már tévednék, ha szívkatéterezésre küldeném őket, és ez orvosilag etikátlan, továbbá gazdaságtalan is. Ha netán mindjárt meg is operálnák őket, ez már súlyos tévedés, műhiba lenne a 120 esetből 40-szer.

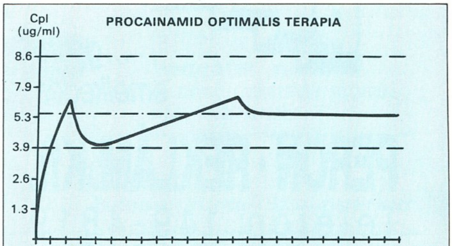
Regionális Kardiológiai Központunk informatikai rendszerének alábbi ismertetése az orvosinformatikai felfogást tükrözi. A jelenleg működő teljes orvosinformatikai rendszer 28 éves kutatómunka és klinikai alkalmazás tapasztalatainak eredménye.

### A rendszer „vérkeringése”

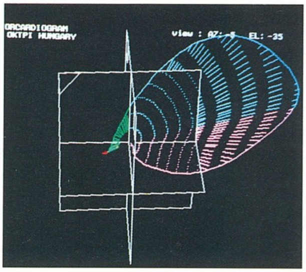
A szöveges adatrögzítésre autocode, optical mark readerrel (OMR) kezelhető adatlapot szerkesztettünk; az amplitúdó-idő függvénygörbéket rendszerelemző eljárásokkal szelektálva 3-4 dimenziós sztereogramokká alakítottuk; a képes információhordozók (röntgen, ultrahang) 2-D elemeiből perspek-




*1. ábra. Egy szívciklus független primer jellemzőinek (áramlás — Q'; nyomás — P; térfogat — Q) változásait tünteti fel egymás és az idő függvényében. Utóbbit a görbe mentén lévő töréspontok jelzik. Ehhez a sztereodiagramhoz bemenetként az amplitúdó-idő függvények szolgáltak.*




*2. ábra. A pont-vonalas egyenes a plazma megadott (kívánt) gyógyszerkoncentrációját mutatja — állandósult állapotban.*




*1. kép.*

tivikus takart térbeli grafikai megjeleníthetőséget biztosító, animáció révén mozgó ábrázolást szolgáltató programokat dolgoztunk ki. Végül a teljes strukturofunkcionális rendszert modellkonstruálás és működtetés segítségével analitikus és integratív módon vizsgáltuk, egyedi optimalizációra alkalmas szimulációkat végeztünk.

Saninform néven egészségügyi memóriakártyar-endszert és eszközt dolgoztunk ki a sürgősségi, alap- és szakellátás adatfelvétele automatizálására. A járó- és fekvőbeteg-ellátást, klinikai kutatást segít számítógépes rendszer 12 programból áll.

Az OMR-adatlapok célzott egészségvizsgálati felhasználását az összesen 127 747 főből álló lakossági reprezentatív minta szolgálta. A páciensenként 50 adatot tetszőleges kombinációkban dolgozza fel a számítógépes rendszer, és ez igen fontos, országos jelentőségű megállapításokhoz vezetett, amelyek egy része az Egészségügyi Világszervezet (WHO) egyik munkacsoportjának is vitaanyagát képezte (például a kisvérköri vérnyomás-fokozódás 4%-os előfordulási gyakoriságát egyedül Magyarországon sikerült megbecsülni). Ugyanezen OMR-lapokon összesítjük a járóbeteg-rendelés orvosi és forgalmi adatait, egy ún. egyéni követéses vizsgálatban — évtizedeken át —, illetve számítógépes összesítő adatfeldolgozást végezhettünk az évenkénti eredmények sorozata alapján.

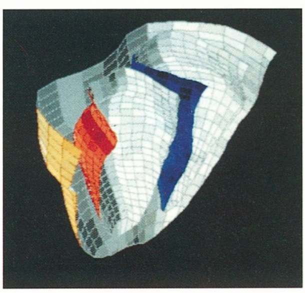
Az OMR-lapok a beteg kórtörténetét és fizikális, valamint egyéb vizsgálati adatait is hordozni tudják. A megfelelő forgatókönyv alapján — kérdőívvel összekötve — részben automatikus zárójelentést készít, részben a vizsgált, kezelt beteg egyéni, tételes költségelszámolását, továbbá az egészségügyi dolgozók teljesítménymérését s ezzel arányos teljesítménybérezését is kiszámítja. A gazdaságossági elemzések fő megállapítása az, hogy a fekvőbeteg-gyógyintézetekben a betegre fordított összköltség 37,11%-át a hotel + energia + diéta + igazgatás, 34,06%-át a vizsgálatok, 0,15%-át a gyógyszerek, 23,26%-át a bérek, és 5,42%-át egyéb költségek teszik ki.

A görbeelemző eljárások közül a sztereó kardiocyclogramot (1. ábra), továbbá sztereó spirogramot és sztereó vektorkardiogramot (1. kép) sikerült kidolgoznunk. Ezek integratív módon tartalmazzák a megfelelő rendszerek állapotának s állapotváltozásának teljes információmennyiségét — szemben a szakirodalom sokféle rendszertelen jellemzőinek halmazával.

A gyógyszerek vérkoncentrációszintjének kiszámítására, a további terápia individuális optimalizálására kidolgoztunk pharmakokinetikai törvényszerűségen alapuló compartment analysist felhasználó modelleket (többféle gyógyszerre), s ennek segítségével számítógépes egyedi gyógyszeradagolást tudunk optimalizálni (például a 2. ábra szerint).

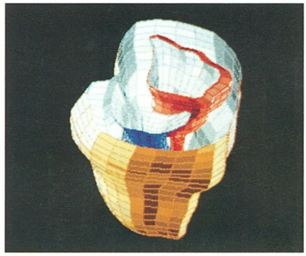
## Látni, hallani, felkészülni a modell alapján

Egyes szív- és érrendszeri betegségcsoportok differenciál diagnosztikájában — a döntés-előkészítésben logikai függvényeket kezelő, illetve a feltételes valószínűség szerinti diganózis-tünet összefüggéseket alkalmazó programokat kidolgozva és felhasználva — az alap-, a komplikatív és a kísérő betegségek előfordulási gyakoriságának megoszlását állapíthattuk meg, mintegy 1000 klinikailag kivizsgált páciens adatainak, jellemzőinek számítógépes feldolgozása alapján. A kóros eltérés nélküli személyek nem és életkor szerinti csoportosítása után klinikai élettani szabályszerűségek váltak felismerhetővé, megfogalmazhatóvá. Így: a kardiopulmonális rendszer normál értékei; a szívműködés frekvenciatörvénye: az emlősök pulzusszáma fordítva arányos testsúlyuk köbgyökével; az emberek alkattípusának életkorfüggése.



*2/a kép*

Az élettani szabályok ismeretében számítógépes modellek megszerkesztésével és működtetésével finom részletű funkcionális vizsgálatokat végeztünk a nagyvérkör, a légzés és a vérgázértékek, a kisvérkör, valamint a szívkoszorúérhálózat tanulmányozására. Ezek legfontosabb és egybehangzó megállapítása az, hogy a szívhez csatlakozó kimenő erek bemenő impedanciája a szívverési frekvencia növelésével csökken. E tör-



*2/b kép*

vény felismerésének élettani, sportélettani, munkaélettani, kórélettani, valamint klinikai jelentősége teljesen új szemléletet formál: a biológia nem a biofizikai törvények ellenében, hanem éppen azok alapján és kiegészítésével működik. Az élő természetesen valóban más, mint az élettelen, de nem ellentétes értelemben, hanem azt tökéletesítő mivoltában.

A számítógépes modellvizsgálatokat a rendszerszemlélet követelményeinek megfelelően a hordozóstruktúrák (maga az emberi test vagy annak részei) elemzésére is ki kellett terjesztenünk. Így a szív ultrahangos, alkalmas módon felvett 2-D képeinek számítógépes feldolgozásával sikerült a mozgó 3-D rekonstrukció, és a szívről alakhű, egyedi térbeli másolatot tudunk készíteni a számítógép képernyőjén kivitelezhető próbaműtétek céljára (2/a és 2/b kép).

Végül mindezek felhasználásával tanácsadó rendszereket képeztünk (például a koszorúérbetegek ellátását segítőt), melyek a magasabb szintű tudás megszerzését, elmélyítését mindenkinek — a szakmában kezdő kollégának is — hihetetlenül meggyorsítja, annak kiteljesítését könnyen lehetővé teszi; természetesen fokozatos tapasztalati kontrollal, megerősítéssel együtt.

**Naszlady Attila**

## Számítógépes kartográfia

# Ami a térképen nem látható

Mi is a térkép? Ez a kérdés talán feleslegesnek tetszik, hiszen amire vonatkozik, az nap mint nap elénk kerül: autóvezetés közben, kiránduláson, külföldi utazáson, vagy éppen iskolapadban ülve földrajzórán; és nagyon valószínű, hogy e fogalom precíz értelmezésével eszünkbe sem jut foglalkozni. Megteszi ezt viszont helyettünk az egyik legérdekesebb szakterület, a térképtudomány, amelynek különleges sajátosságai mellett van egy napjainkban tipikus vonása is: a számítástechnika alkalmazásának gyors térhódítása.

Úgy érzem, nekem mint térképésznek félre kell tennem a kötelező szakmai elfogódottságot, és be kell vallanom, hogy igaza van mindenkinek, aki szerint a térkép egyszerűen csak használati cikk, egy a boltokban megvásárolható ezernyi közül.

Hogy mit értsünk viszont térképtudományon, már nem ilyen egyszerű dolog. Mindig az adott publikáció szerzőjének szakmai „terjeszkedési vágyától" függ: van, aki ide sorolja például a földméréstant (geodéziát), más a távérzékeléssel kapcsolatos ismereteket, van viszont, aki ezeket saját lábukra hagyja állni. Én ez utóbbi véleménnyel értek egyet, így számos témát csak érinteni fogok, melyek az említett „vitatott területeken" vannak.

Visszatérve a térképre, adható egy rendkívül általános meghatározás: „A térkép térbeli információk megjelenítése speciális szempontok szerint, meghatározott szimbólumrendszer segítségével". Az olvasó kezébe kerülő térkép valószínűleg szűkebb definícióval is leírható, hiszen a mindennapi életben csak a tájékozódást segítő, a földfelszín egy-egy darabját bemutató, papírra nyomott kiadvánnyal találkozunk.

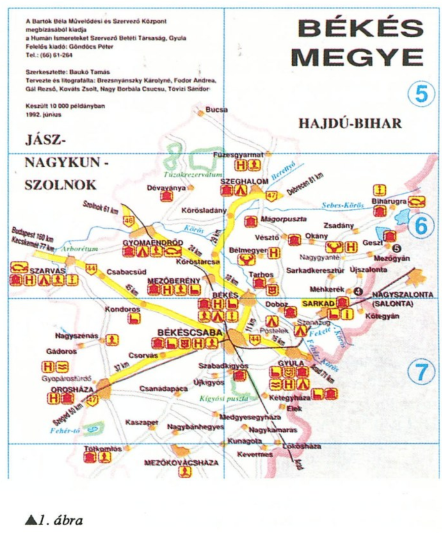
De térkép a csillagtérkép, az űrfelvételek alapján készült átdolgozott fotómontázs, és a (jobb) gépkocsik fedélzeti számítógépén megjelenő tájékozódási vázlat is. Talán a legfontosabb közös pont az a bizonyos nehezen megfogható szimbólumrendszer: ha ön ránéz a térképre, tudja, hogy két párhuzamos vonal valószínűleg egy utca rajza, és a beleírt betűk az utca nevét adják. Ez triviálisnak tűnik, mégis évszázadokig tartó kemény, és gyakran igazán (nevében is) tudományos kartográfusi munka kellett ahhoz, hogy mindez tényleg magától értetődő lehessen. Mi, mai térképészek már beleszülettünk a készbe, ismerjük a térképhasználók igényeit, szakterületünk gyakorlatilag minden indokolt fejlesztési tevékenysége a technológia korszerűsítése érdekében történik.

Tehát a „fogyasztó" még egy jó ideig semmit sem fog észrevenni, nem fogjuk felkavarni a térképről kialakított elképzeléseit. Mert az, hogy a térkép szebb és információgazdagabb (esetleg olcsóbb) lesz, még semmilyen változást nem jelent, csak szükségszerűség, amit ráadásul joggal vár el tőlünk. Persze a számítógépek és az audiovizuális médiumok uralma arra késztetnek minket is, hogy lényeges előrelépésen törjük a fejünket.

Egy lehetséges irány: a hagyományos, papírra nyomott térképek az információk egy adott (térbeli és időbeli) csoportját mutathatják, míg a számítógép képernyőjén könnyen megjeleníthetők bizonyos információkapcsolatok. Például Budapest utcahálózatára rávetíthetők a legfrissebb forgalmi adatok, a gép kiszámítja az optimális útvonalat, és azt javasolja. Ha ez így megvalósul, az egész rendszert továbbra is térképnek hívjuk majd, és mégis mennyire más ez a fogalom, mint amiből kiindultunk!

### Honnan és mi kerül a térképre

A térkép „térbeli információkat" mutat be, kérdés, hogy ezen információk milyen formában állnak rendelkezésre. A Föld felszínének jelentős része évezredek során hagyományos földmérési módszerek révén vált térképen ábrázolhatóvá. Ezek az információk természetesen analóg jellegűek, rajzokon, nyomtatott térképeken találhatók, és természetesen számtalan könyvben, statisztikában vannak meg azok az adatok, melyek ábrázolása a térképen szükséges és lehetséges. A modern térkép-




▲*1. ábra*

készítő is ilyen adatokra támaszkodik. (Lásd például az 1. ábrát, Békés megyét, ahol a térképész az idegenforgalmi szempontból lényeges adatok alapján tüntette föl a szolgáltatásokra, látnivalókra utaló jeleket.)

Minőségi változás két területen következett be. Egyrészt megjelentek azok az eszközök, melyek segítségével az információk a felszínről gyakorlatilag közvetlenül digitális formában nyerhetők, valamint a számítógépek elterjedésével mind több kiegészítő adat lett korszerű módon feldolgozva és tárolva. (Persze fontos megjegyezni, hogy ez utóbbi anyagok — például egy lakosságstatisztika — csak a térképész számára tekinthetők már digitális, könnyen kezelhető forrásnak, maga az adatgyűjtés egy kicsit nehézkesebb és „hagyományosabb”.)

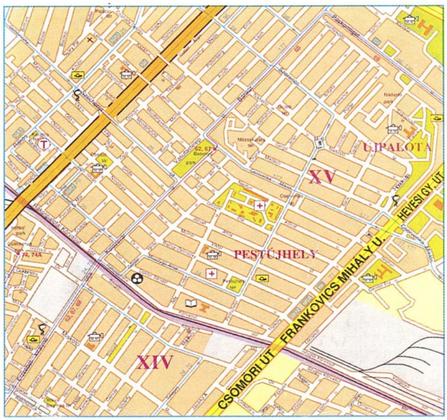
## „Nincs műszer, mellyel mindez jól megmutatható”

A Föld számos helyén és sok tudományhoz kapcsolódva óriási jelentősége van a korszerű távérzékelésnek és ezen belül az űrfelvételek feldolgozásának. Döntő érv emellett, hogy az érzékelőrendszerek egy csoportja digitális „képet” ad, amelynek feldolgozási lehetőségei széles körűek, és a különböző hardver- és szoftvereszközök legmagasabb szintjét használják. Nagy területű, valamint a kevésbé feltárt országokban tájékozódást segítő térképek közvetlen alapjául is szolgálnak. Magyarországon, és általában Európában közel sem lehet ekkora szerepük, ugyanis ez a kontinens gyakorlatilag nagy részletességgel térképezett, és az űrfelvételek — legalábbis ilyen szempontból — semmi újat nem tudnak mondani.

### Programcsomagok

Egy kevésbé ismert térképészeti program a svájci OCAD, melynek kezdeti változatai tájfutó térképekre „szakosodtak”, de rövidesen megjelenik az általánosított mutáció is, ebben mi alakíthatjuk ki a rajzi jelkulcsot. Vannak olyan programok, amelyekhez térképészetet támogató modulok szerezhetők be. Ilyen általános programcsomag például az Intergraph rendszer, ennek létezik minigépes és PC-s változata is. Sokan ezt tartják a jelenlegi legjobb megoldásnak. A „verseny” egy másik indulója az AutoCAD, kicsit szerényebb, de jól használható programcsomaggal. És természetesen van még több hasonló program, amelyekről részletesen lehetett olvasni az Alaplapban is.

A másik gond, amit a nagyfelbontású űr- és légifelvételek odaadó hívei elfelejtenek: ezeken a képeken ugyan látszik, hogy hol van az utca, de az nem, hogy mi a neve; az látszik, hogy hol van az épület, de az nem, hogy milyen intézmény van abban az épületben. Az sem valószínű, hogy a műholdról jött digitális kép tartalmazni fogja azt, hogy merre jár a hetes busz.




▲2. ábra

Még egy érdekes probléma: bár például a pontos út- és utcaszélességek könnyen rögzíthetők, de ezt így csak a térképek szűk köre (például a topográfiai térképek) ábrázolja, az „egyszerű térképek” nem a valódi szélességet ábrázolják, hanem inkább a fontosságot. (2. ábra).

A térképen ábrázolt információk jelentős része ebbe a nem látható, (néha elvont) csoportba tartozik. (Gondoljuk meg, hogy a békési térképhez a sárga szín ún. gépanyaga — lásd a 3. ábrát — tartalmazza például a „Túzokrezervátum” feliratot is, és a terület kontúrját. Mi mindenre tudunk pusztán ennyiből is következtetni!) Ezekből egy rész különböző adatbázisokban rögzíthető (lakosságadatok, útminőségadatok, intézménylisták, utcanévjegyzékek stb.), de mindig marad egy tetemes rész, melyet a térképszerkesztő tudása képvisel. A grafikai alap (az utcák, épületek elhelyezkedése stb.) a nagyközönségnek szóló térképeknél az előző kiadás nyomata: a jó minőségben rendelkezésre álló részletes alaptérképek, azaz elsősorban nyomtatott, nem pedig digitális anyag a bázis. (Itt — talán nem későn — hívom fel a figyelmet arra, hogy a cikk megállapításai elsősorban az általánosan használt autó-, város-, turista- stb. térképekre igazak, egy részletesebb, például kataszteri térképnél már más preferenciák vannak, de ez egy másik cikk témája lehet...)

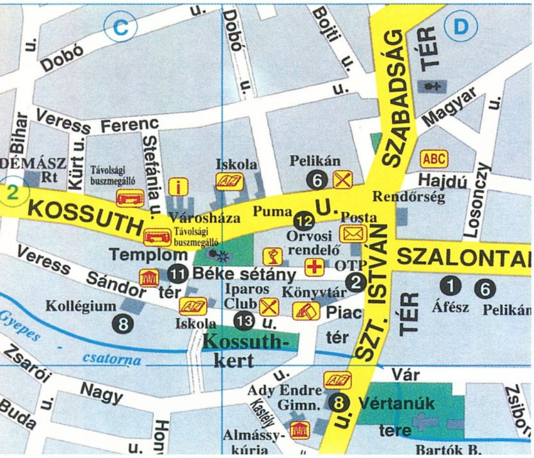
## A térinformatikai rendszer

Kérdés az, hogy milyen lehetőségeink vannak az említett anyagok beépítésére egy olyan rendszerbe, melynek végső



▲3. ábra

▼*4. ábra*




outputja a kész térkép. A legkorrektebb válasz, hogy dolgozzunk ki egy teljes ún. térinformatikai rendszert, mely tartalmazza a szükséges hardvert és szoftvert is, a bevitt adatokkal együtt. E probléma megoldása egy rohamosan fejlődő új tudományág feladata, mely átfedésben van a térképészettel, de annál bizonyos értelemben (technikailag) sokkal szélesebb területű, és idővel szükségszerűen „le fogja nyelni" a kartográfiát, mint egy — a globális rendszer egy kimenetén dolgozgató — szakágat. A „kisebb testvér" szemszögéből talán érthető, hogy az előnyök mellett a korlátokról is szólok.

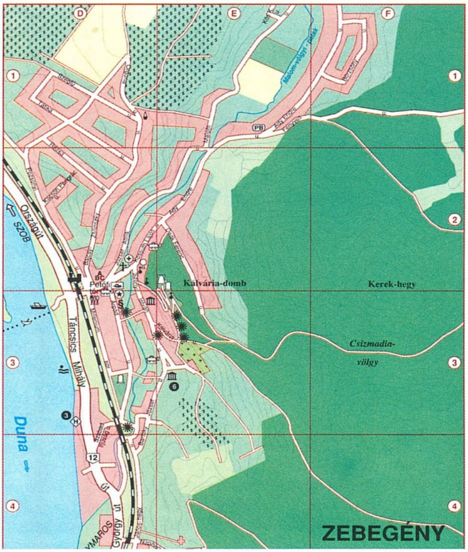
A „térinformációs rendszer" is tág fogalom, számunkra elsősorban a tartalmi különbségek lényegesek. Léteznek ún. digitális domborzatmodellek, amelyeknek előnye, hogy nem szorulnak folyamatos javításra, és a térképeken használt különböző domborzatábrázolási módszerek (szintvonalazás, árnyékolás) ebből számítógép segítségével levezethetők. Másik jól elkülöníthető elemcsoport az ún. síkrajzi modell, mely a felszínen található egyéb objektumokat (út, vasút, víz, település stb.) tartalmazza digitális formában. Általában jellemző, hogy az egyes országok részletes topográfiai térképeiket digitalizálva alakítják ki az ilyen információs rendszereket. Az adatok óriási mennyisége miatt ez kifejezetten nagygépes feladat — legalábbis egy részletes adathalmaz teljes összefogása esetén. Több nyugati országban gyakorlatilag teljesen lefedték a területet ilyen digitális térképekkel (például USA, Nagy-Britannia), de Magyarországon is elkészültek, illetve készülőfélben vannak ilyen munkák. Feltétlenül meg kell említeni a Geometria ilyen irányú tevékenységét: egyrészt Budapestről, másrészt az egész országról készítettek és forgalmaznak komplex térbeli adatbázisokat.

Általában jellemző ezekre a rendszerekre, hogy igazodnak a megrendelők speciális igényeihez a tematika és az ábrázolt terület tekintetében, hiszen minden felhasználó számára megfelelő általános rendszert képtelenség kialakítani. A térképészek mint grafikai alapot jól használhatják ezen rendszerek szolgáltatásait, de itt általában be is fejeződnek a lehetőségek. Vizsgáljuk meg ennek elsődleges okait!

— A térképeknél alapvető ismérvnek kell lennie a naprakészségnek. Ezt elvileg támogatja a számítógép alkalmazása, viszont információkezelési kultúráról beszélni Magyarországon még nemigen lehet, így az egyszer valahogy létrehozott digitális rendszer egy év múlva rohamosan veszít értékéből — az adatkarbantartók legjobb akarata ellenére is.

— Mivel egyetlen magyar térképész cég sincs olyan tőkeerős, hogy saját maga befolyásolhassa a készülő rendszereket, így valószínűleg egy ideig még nem kaphatjuk azt, amit szeretnénk. Ez a helyzet arra mutat, hogy az említett cégek saját maguk alakítanak ki ilyen-olyan ideiglenes „rend-




◀*5. ábra* ▲*6. ábra*

## Színek és rétegek

AutoCAD-del a térképészeti feladatok túlnyomó része megoldható, legfeljebb kényelmi problémák vannak. Gond az elemi alaplépések sorozatos végrehajtására kényszerítő logikája — a térképek kialakításánál gyakran lenne praktikusabb, ha összefűzött műveletsorokkal tudnánk dolgozni, például, ha egy utcát rögtön kétvonalasra rajzolhatnánk.

Előny az AutoCAD rétegstruktúrára épülő rendszere, mely tökéletesen megfelel a jól ismert és begyakorolt hagyományos techológiai lépéseknek. A térképkészítés során ugyanis mindig a végső fázist, a nyomtatást tartjuk szem előtt. Az eddigi, fotózási lépésekre épülő kidolgozás igényelte az egyes színekhez, és azon belül az egyes raszterfokozatokhoz tartozó térképi elemcsoportok külön kezelését. Ez az elv most is megmarad, és ezt könnyen követni tudjuk a rétegek jó beosztásával. Ahogy korábban a szerkesztő végiggondolta, hogy mit hova kell rajzolni ahhoz, hogy a kívánt színben és raszterben jelenjen meg, most rétegstruktúrát kell terveznie. Nyilvánvalóan külön rétegre kerülnek a folyók (kékkel) és a településnevek (feketével).

Ezután a térkép megrajzolása már „gyerekjáték", kihasználhatjuk az összes rendelkezésre álló grafikai parancsot, melyeket nincs értelme itt részletezni — hiszen ez a folyamat nem sokban különbözik bármely grafikus programban megismerhetőtől. Ezen okból is sokan előnyben részesítenek bizonyos grafikai programokat (például CorelDraw), melyekkel a térkép végül is megrajzolható, több szempontból szebben és egyszerűbben, mint egy CAD programmal. Viszont nagyon hiányzik az említett rétegstruktúra, és komoly méretkorlátok is vannak.

A CorelDraw programban egyszerűen és követhetően megoldható a színbontás, és például a postscript formátum kialakítása, melyet már fogadni tud a levilágító. Az AutoCAD-rajzot az ún. Autoscript program segítségével dolgozzuk át, mely a .DXF fájlt .PS-re konvertálja (ez a program Európában kevésbé ismert). Ebben a programban rendeljük hozzá az egyes AutoCAD-rétegekhez, pontosabban színekhez a megfelelő paramétereket. Legfontosabbak: az adott rétegen (színben) található elemek vonalvastagsága és raszterértéke. Ez a két adat (más fontos kiegészítésekkel együtt) megadja az adott elem levilágításának mikéntjét; természetesen a betűknél a betűtípus lép a vonalvastagság helyére.

szereket", minden egyeztetés nélkül, az éppen készítendő térképre koncentrálva (a korrekt adatbázisok létrehozási és fenntartási nehézségei miatt itt rendszeren csak az ún. grafikai alapot értjük, célorientált szoftver-hardver támogatással). Érdekes lehet, hogy számos nyugati cégnél is ez a gyakorlat (például Freytag-Wien), csak legfeljebb mindezt gyorsabban és szervezettebben teszik.

— Az országos rendszerek általában egy vagy két méretarányban készülnek el, egy részletesebb digitális alap átdolgozásánál ún. generalizálásra van szükség, ami a hagyományos kartográfiának is talán legproblematikusabb része. Például rendelkezésre áll egy 1:100 000 méretarányú anyag, de nekünk 1:500 000-es térképre van szükségünk. Logikus, hogy ez korántsem oldható meg egyszerű kicsinyítéssel, hanem szűrnünk kell az információkat. Az, hogy a nagyobb léptékű térképen lévő 10 faluból melyik az az egy, amelyik megmarad a kisebb léptékűn, gyakorlatilag szubjektív (rengeteg ismeretet igénylő) döntés eredménye. (Ez olyan pont, ahol meghal a számítógép racionális és véges „agya".) Azaz valószínűleg úgy döntünk, hogy a nulláról kezdünk a térképünkkel, illetve keresünk egy hasonló méretarányú, nyomtatott kiadványt.

### Művészi igényesség és technikai kompromisszum

Ezek alapján mi napjaink gyakorlata? A már meglévő kiadványokból digitalizáljuk a szükséges elemeket. Eszközünk gyakorlatilag kizárólag a digitalizáló tábla, szkennert többek között azért sem használunk, mert az „ész nélkül" mindent bevisz a rendszerbe, míg nálunk a digitalizálás közben erős szelektálást, generalizálást kell végeznünk (annyira, hogy ez a munka is kifejezetten térképszerkesztői ismereteket igényel). Tapasztalataink szerint a digitalizálás egyébként is időben elenyésző részt tesz ki a teljes feldolgozásban, így kevés a késztetés, hogy annak automatizálására törekedjünk. Más lesz a helyzet akkor, mikor szükségessé válik a nagyméretű, már filmen lévő munkák bevitele a gépbe.

A kiegészítő adatokról több adatbázis áll rendelkezésre, a térképen található nevek (például a 4. ábrán Sarkad belvárosa, a fekete szín gépanyagai) már régebben a gépbe kerültek, aminek az automatikus fényszedés előnyei mellett az is a célja, hogy más kiadványoknál felhasználhatók legyenek.

Nem felejthetjük el, hogy minket elsősorban a végtermék tartalma, minősége, szépsége érdekel, hogy azt alakítsuk olyanná, ami mindenki könnyen és jól tud használni. A rendszert ennek megfelelően kell kiválasztanunk, olyan szoftverre és hardverre van szükségünk, amely mindezt gazdaságosan biztosítja.

Nézzük először a hardvert. Egy igényes kivitelű térkép rengeteg elemet tartalmaz. Sok kiadványhoz nehéz lenne olyan gépet találni, amelyikkel jól kezelhetők lennének. Nem is a tárolókapacitás jelent akadályt, hanem a gépek relatív lassúsága, így számunkra különösen fontos minden olyan technikai előrelépés, mely lényegesen növeli a sebességet. (Ilyen lehetőség például az ún. sejtprocesszor alkalmazása.)

### A kartográfia és a PC

Termelési körülmények között persze mindig csak a technika második vonala található — igaz, kérdéses, hogy mit értünk ezen. Kétségtelen, hogy az említettek miatt célszerű egy minigépes környezetre építeni. Ezt célozza meg a számítógépes térképészet egyik fő rendszerszállítója, az Intergraph cég; Magyarországon is több helyen működik munkaállomásuk, például a Carto-Hansánál is, ahol német megrendelésre készülnek digitális — többek között fotogrammetriai — anyagok. Bár nagy a kísértés, ennek ellenére a ténylegesen piacra termelő cégek megmaradtak az IBM PC-knél, illetőleg a Macintosh gépeknél, de főleg az előbbinél. Ennek oka a PC-k elérhetősége, elterjedtsége és kedvező teljesítmény/ár hányadosa.

Más szakterületekhez hasonlóan nálunk is a térképész kénytelen megtanulni az alapvető számítástechnikát, és nem a programozó a kartográfiát. Azaz kifejezetten nyitott, jól kezelhető rendszerre van szükségünk, melyet az is könnyedén tud használni, aki egy héttel előtte még nem is látott számítógépet. Ezt a lehetőséget jelenleg csak a PC-k nyújtják, természetesen az óriási szoftverválasztékkal együtt. PC-kategórián belül bizonyos minimális követelményeket fel kell állítanunk, a feldolgozandó térkép méretétől (effektív méret és tartalmi sűrűség) függően. Egy A2-es térképhez elegendő egy 25 MHz-es 386-os gép, de egy B1-eshez már 486-os kell, lehetőleg nagy RAM-diszkkel. A VGA monitor és az egér elengedhetetlen, és legalább a munka elejére szükséges például egy A3-as digitalizáló tábla is. Ilyen paraméterekkel már

neki lehet vágni a nagy feladatnak, térképet csinálni számítógéppel.

Kifejezetten térképész szoftver azonban igen kevés van, hiszen ez a piac nagyon szűk. Az egyáltalán alkalmazható szoftverek választéka sem akkora, mint szeretnénk. (Lásd a keretes anyagot.) A térképészek kicsit tanácstalanok, sajnos inkább azért, hogy melyik a kevésbé tökéletlen program... (Biztosan sokan vitába szállnak velem, hogy miért az AutoCAD-re szavazok, de én ezt érzem igazán „közös nyelvnek", hasonlóan, mint az adatbázis-kezelők közt a dBase-t, a gépek közt az PC-t.)

Mindenesetre ezekkel a szoftverekkel meg tudjuk csinálni a feladat tárgyát képező térség képét — a munka lépései, változatai és eredménye előttünk alakulnak a monitoron.

## A térkép „megfoghatóvá válik"

A következő igazán fontos lépés már ahhoz szükséges, hogy ne csak a monitoron gyönyörködhessünk alkotásunkban... Ami a monitoron látható, már térkép ugyan, de nem olyan formában, ahogyan azt a mindennapi életben használjuk. Noha számos más módja van a megjelenítésnek, most csak azt az utat járjuk végig, ahogy a papírra nyomott térkép készül.

A céltól visszafelé kell elindulnunk: annyi nyomdakész filmre van szükségünk, ahány színű a térkép. (Például az 5. ábrán látható térképhez nyolcra. E filmek előállítására lézeres levilágító berendezések szolgálnak — amelyekhez az inputot kell biztosítanunk. A felhasznált szoftverektől függ, hogy milyen lépésekre van ehhez szükség.

A térképészetben mindig szükség van ún. próbanyomatra (például ilyen az 6. ábra eredetije is — Zebegény), ez megoldható egy lézerprinteren, vagy szoftver úton postscript fájl fogadására képessé tett mátrixnyomtatón. Az ellenőrzések és javítások után sor kerülhet a levilágításra. Nem véletlen, hogy ragaszkodunk ehhez a kimeneti eszközhöz, ugyanis csak ez biztosítja a nyomtatáshoz szükséges minőséget. (Térképet elő lehet állítani plotteren is, de az a térkép csak másfajta igényeknek felel meg.)

A levilágító paraméterei jelentik a következő szűk keresztmetszetet, Magyarországon jelenleg legjobb tudomásom szerint A2-es méretű film a nyerhető legnagyobb formátum. Természetesen lehetőség van nagyításra, és ez kb. B1-es méretig meg is oldható. Valószínűleg párhuzamos folyamat lesz az, hogy mi egyre nagyobb térképeket tudunk rajzolni, és közben megjelennek (pontosabban elérhetők lesznek) az ezeket fogadni képes levilágítók.

**Kováts Zsolt**

*(„Társszerzők", továbbá a mellékelt illusztrációk által reprezentált munka nagy részének végzői: a rendszert kidolgozó és betanító A. J. Kimerling (Oregon State University) és Puskás János komputer-kartográfiai szakértő.)*

## Intelligens ipari tanácsadó rendszerek

# A „pokoli tornyok" megfékezése

**Századunkban az egyre többet tudó automatizált ipari rendszerek széles körű elterjedésének vagyunk tanúi. Azonban minél komplexebb egy rendszer, annál kényesebbek a benne zajló folyamatok és ezek kölcsönhatásai, annál bonyolultabb és nehezebben kezelhető hibajelenségek léphetnek fel a rendszerben. Ráadásul ezek hatása bizonyos esetekben katasztrófához vezethet... (például atomerőművekben, vagy veszélyes anyagokat előállító vegyiüzemeknél).**

Még a legmagasabb automatizáltsági fokú ipari rendszer működtetéséhez is szükség van emberi kezelőkre, akik a technológia működését jól ismerik, a benne lezajló változásokat folyamatosan követik, és ha szükséges beavatkoznak, hogy a technológia a lehető leghatékonyabban működjön, az esetleges meghibásodások hatása a lehető legkisebb legyen.

A kezelők hatalmas anyagi és erkölcsi felelősségére az egész világ közvéleménye felfigyelt a csernobili atomerőmű robbanása után, de ehhez hasonló esetek a világ biztonságosabbnak tartott felén is előfordultak, ahogy ezt a „Kína-szindróma" című filmben végigizgulhattuk, és ahogy a valóságban például a Three Mile Island amerikai atomerőműben 1978-ban megtörtént.

Ott a központi reaktormag elsődleges hűtőrendszerében csőtörés következett be. A biztonsági hűtőrendszer automatikusan bekapcsolt ugyan, tehát nem lett volna komolyabb baj, de az operátori vezénylőterembe néhány perc alatt több mint száz különböző vészjelzés futott be. A hatalmas információtömeget a kezelők nem voltak képesek helyesen értelmezni, és egy hibás döntés folytán lekapcsolták a biztonsági hűtőrendszert. Ennek hatására a reaktormag megolvadt; csak kis híja volt, hogy a dolog igazi katasztrófához nem vezetett.

Az ilyen és ehhez hasonló esetek vetik föl azt a kérdést, hogy ha a kezelő felelőssége ekkora, vajon megvannak-e hozzá a szükséges eszközei, hogy jól és gyorsan döntsön kritikus helyzetekben? Bár a legtöbb ipari rendszerben a döntés súlya kisebb, mint a felsorolt példákban, a válasz mégis egyértelmű: a kezelő általában nem kap elég támogatást döntéseinek meghozatalakor. Éppen ezen a problémán segítenek az intelligens számítógépes felügyelő- és tanácsadó rendszerek, amelyekben több tudományterület kutatásai/eredményei vannak beépítve.

### Csak egyszerűnek tűnik...

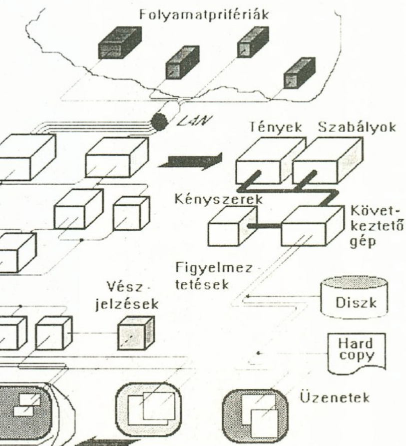
Bármely kezelői döntés megfogalmazható feltétel—akció párokból álló, hagyományos (ha—akkor típusú) szabályok formájában. (Például: „Ha a kazánban túl magas a nyomás, és az automata biztonsági szelep nem nyit ki, akkor csökkentsd a nyomást a kéziszelep segítségével.") Ezeknek a szabályoknak a feltétel részében általában vészjelzések vagy hibajelzések adott kombinációja szerepel, akció részük pedig a beavatkozás (tehát a vészállapot elhárításának) módját tartalmazza.

Ettől a ponttól kezdve a helyzet látszólag egyszerű: kövessük a hagyományos receptet, azaz keressük meg az adott technológia legjobb szakértőit, faggassuk ki őket, hogy mit tennének, ha ez vagy az történne — és ezzel lényegében meg is van a szabálybázis, mely a lehető legjobb minőségű döntéseket garantálja. Az élet azonban ennél sajnos egy kissé bonyolultabb.

### Időállandók és nemmonoton következtetések

Rendkívül lényeges a tanácsadó rendszer válaszideje, azaz hogy egy adott esemény (például vészjelzés) bekövetkezése után mennyi idő kell a tanács generálásához. Ezt a felügyelt rendszer időállandói (tulajdonképpen a benne lejátszódó változások sebessége) szabják meg, ami egy — a válaszidőre adott,

A REALEX információ feldolgozási sémája



fix — időkorlátot jelent. Ha a tanács ezen az időkorláton túl érkezik, már értelmetlen (mert például már fölrobbant az atomerőmű). Ez röviden és számítástechnikai terminológiában úgy fogalmazható meg, hogy tanácsadó rendszerünktől valós idejű működést, várunk el.

A második probléma egy ellentmondás feloldása: a tanácsadó rendszer egyrészt valós időben működik hosszú időn keresztül, másrészt azokat az információkat, amelyekre a következtetéseit alapozza, közvetlenül vagy számítással származtatva a technológián végzett folyamatos és automatikus mérések eredményeiből szerzi — viszont a technológia állapota az időben szüntelenül változik. Ezért a tanácsadó rendszernek képesnek kell lennie folytonosan alkalmazkodnia a változó környezethez, azaz korábban levont következtetéseit visszavonni és helyettük újakat hozni. Ezt a tulajdonságot nemmonoton következtetésnek nevezik.

## Modellbázisú rendszerek

A harmadik problémát az jelenti, hogy mivel a technológiát ismerő szakértők különböző rétegeinek tudása jelentősen eltérő vonásokat mutat, ezért a szabálybázisú tudásleírási mód nem minden esetben megfelelő. Ha a szakértőket három fő csoportba soroljuk, mint tervezők, technológusok és kezelők azt tapasztaljuk, hogy a tervezők leginkább matematikai modellekben gondolkodnak, a kezelők lényegében feltétel—akció párokban, a technológusok pedig e kettő keverékében, gyakran kvalitatív módon, például strukturális kölcsönhatásokban és állapotátmenetekben. Mindez arra ösztönöz, hogy a tanácsadó rendszerbe az egyszerű szabályalapú megközelítés mellett többszintű modelleket építsünk be. Ez egyébként az ún. modellbázisú rendszerek alapgondolata is, melyeket a szakirodalomban második generációs szakértői rendszerekként is szoktak emlegetni.

## A Realex keretrendszer

Az eddigiekben elemzett problémák megoldására és intelligens, valós idejű ipari tanácsadó rendszerek létrehozásának támogatására született egy megoldás a BME Műszer- és Méréstechnika Tanszékén.

A Realexben a felügyelni kívánt technológiáról többféle információt kell megadni, amikor egy dedikált (azaz egy adott technológiához kifejlesztett) intelligens felügyelőrendszert építünk *(ábra)*. A matematikai modell a technológiáról érkező jelek elsődleges értelmezését és feldolgozását végzi objektumorientált feldolgozó modulok megfelelő hálózatba kapcsolásával. A kvalitatív modell a technológia egyes strukturális elemei között meglévő kényszerkapcsolatok leírását teszi lehetővé, és az ún. tapasztalati modell szabályainak feltételrészeiben felhasználhatóvá. A valós idejű és nemmonoton működés érdekében a szabálybázist ún. adatfüggőségi gráffá kell fordítani, így elkerülhető a következtetés során a mintaillesztés, és az alkalmazható szabályok keresése. A technológia egyes paramétereinek értéke, illetve az alacsony szintű vészjelzések a technológiai folyamat sematikus ábráit szemléltető sémaképeken jeleníthetők meg, melyeket szintén a rendszerépítés során kell megszerkeszteni.

A Realex első verziója 1987 őszén készült el, azóta több gyakorlati alkalmazása is bevált, melyek közül a legutolsót hozzuk példaként.

## Mikrohullámú átjátszólánc

Az alapprobléma az olyan távközlési hálózatok esete, melyek reléállomásokból állnak; ezek különböző típusú átviteli csatornák (például televízió, rádió, telefonvonalak, telexvonalak) nagyobb távolságra való átjátszására alkalmasak. A reléállomások egymástól 20-30 km-re helyezkednek el, gyakran lakott területen kívül, és mert nagyon megbízhatóak, nincs értelme minden állomáson külön kezelőszemélyzetet tartani. Ezért egy-egy központi helyen akár több száz reléállomást felügyelnek.

A kezelő feladata az esetleges meghibásodások helyének behatárolása, és a szerelők kiküldése az adott állomásra. A gyors javítás nagyon fontos, mert a reléhálózat által nyújtott szolgáltatás rendkívül drága, így kiesésekor az üzemeltető (például a Magyar Posta) a bérlőknek (például a Magyar Televíziónak) tetemes kártérítéssel tartozik.

A rendszer tetszőleges felépítésű és topológiájú reléhálózatok felügyeletére alkalmas egy erre a célra kialakított konfiguráló környezet segítségével. MS Windows alatt működik, ennek megfelelően kezelése egyszerű és hatékony. A reléhálózat struktúrája, továbbá az egyes reléállomások belső szerkezete alapján, valamint pillanatnyilag 100 szabály segítségével képes a hibás reléállomás, azon belül a hibát okozó áramköri kártya automatikus meghatározására (és ezen információknak a kiírására a képernyőn).

**Tilly Károly**

# Egy fejlődésneurológiai szakértő rendszer
# Diagnózis egyszerű és „cifra" esetekre

Jó diagnózis nélkül nem képzelhető el érdemi terápia. Ez korokon át főként egy-egy, a beteg környezetében praktizáló ember, a „gyógyító" — koronként és helyszínenként: a sámán, a kuruzsló, a javasasszony, a bába, a felcser (és a többiek...), legjobb esetben a doktor — rátermettségén, iskolázottságán, tapasztalatain és „hetedik érzékén" múlott. Napjainkban már más (is) a helyzet.

Az MTA SZTAKI Intelligens Rendszerek Csoportja és a Szabadsá-hegyi Gyermekgyógyintézet Fejlődésneurológiai és Neurohabilitációs Osztálya közös célul tűzte ki a csecsemőkori idegrendszeri sérülések korai felismerésére és a kezelés optimális meghatározására szolgáló szakértő rendszer kifejlesztését.

A Vámos Tibor akadémikus irányításával kutató MI-szakemberek és a Katona Ferenc professzor vezetésével részt vevő orvosok együttműködésének eredményeként létrejött egy általános tudásalapú keretrendszer (KAS — Knowledge Application Shell).

A KAS három, egymáshoz kapcsolódó és együttműködő modulból épül fel:

— Szemantikusan szervezett, esetérzékeny adatbeviteli program, amely előre megadott szabályok dinamikus kiértékelése révén a lehetséges kérdések gráfjából a konkrét esetnek megfelelő bejárási utakat választja ki. Segítségével az adatokat gyorsan és kényelmesen lehet a számítógépbe juttatni, onnan bármikor könnyen vissza lehet keresni, módosítani vagy kiegészíteni. A felhasználó az adatokat a programmal folytatott párbeszéd során rögzíti.

— Felhasználóbarát tudásbázis-kezelő program, melynek segítségével — számítástechnikai szakember közreműködése nélkül — intelligens, tudásalapú adatrögzítő rendszereket lehet létrehozni és karbantartani, a felhasználói igényeknek megfelelően.

— Tudásalapú lekérdező program, amely összetett feltételrendszerek (lekérdező jobok) kiértékelésével az adatbázis statikus és dinamikus elemzésére, hipotézisek ellenőrzésére nyújt lehetőséget.

## „A gyermekekért"

A KAS programcsomag alkalmazása egy fejlődésneurológiai tudásbázissal feltöltött rendszer (NES — Neurodevelopmental Expert System), amely sérült idegrendszerű újszülöttek gyógyításával foglalkozó orvoscsoport tudását reprezentálja. A NES tudásbázisa a betegek azonosító adatain kívül tartalmazza mindazokat a vizsgálatokat és kórelőzményeket (terhesség és születés körüli események, előzőleg más intézményben elvégzett vizsgálatok eredményei), melyek a diagnózis felállításához szükségesek, valamint az alkalmazható terápiákat. A tudásbázis részét képezik azok a szabályok is, melyek segítségével a program az összes lehetséges vizsgálat és terápia halmazából az adott esetnek leginkább megfelelő részhalmazokat választja ki.

Ez a rendszer évek óta segíti a Fejlődésneurológiai Osztály orvosainak napi ambuláns tevékenységét is azzal, hogy sok papírmunkát helyettesít, statisztikai adatokat szolgáltat, valamint ez a rendszer az alapja és eszköze a tulajdonképpeni kutatási program megvalósításának, a diagnóziskészítésnek. A kutatás fő iránya, hogy a NES tudás- és adatbázisán olyan, mintaillesztésen alapuló diagnosztikai rendszert építsen, amely a probléma összetettségének modellezésében, az eredetét és mértékét tekintve sokféle, az adatgyűjtés során dinamikusan változó bizonytalanságok kezelésében a természetes orvosi gondolkodáshoz (szindrómák, Gestalt-elmélet) közelebb áll, mint egyéb MI-technikák.

A diagnózist ez a rendszer úgy készíti, hogy kiválasztja a betegségeket tipikus formájukban leíró minták közül azokat, amelyek egy megadott metrika szerint a legközelebb állnak a beteg vizsgálatieredmény-halmazához.

Egy minta az adott betegség (szindróma) tényét és/vagy jellegét, esetleg súlyosságát meghatározó tüneteket (szimptómákat), továbbá a betegséget valószínűsítő kórelőzményeket, valamint a betegség gyógyításához javasolt gyógyszeres és egyéb kezeléseket, terápiákat tartalmazza.

A szakterület sokszínűségét mutatja, hogy a már elkészült minták nagyon sokfélék. Akadnak közöttük néhány feltételt (vizsgálatot) tartalmazó, és vannak nagyon összetett, sok feltételt és azok közötti bonyolult diszjunktív és konjunktív kapcsolatokat tükröző minták: ahogyan vannak viszonylag könnyen, egyértelműen megállapítható bajok — ilyenek ezen a területen az egyes motoros és szenzoros károsodások —, míg vannak „cifra", csak sok vizsgálat után, sok összefüggés elemzésével diagnosztizálható betegségek — például az értelmi károsodás.

## A minták szerkezete

Az orvos a beteg vizsgálata során elemi információkhoz jut (kórelőzmények, laboratóriumi leletek, közvetlenül elvégzett fizikai vizsgálatok stb.). Ezeket az elemi információkat a betegség tényének és jellegének megállapításához szükséges feltételek formájában a diagnózisminták sorai reprezentálják. A feltételek, vizsgálati kimenetelek a betegség meghatározásában, illetve leírásában természetesen különböző szerepet játszanak. Ennek megfelelően ezek — tehát a diagnózismintát felépítő feltételek — egymástól eltérő funkciójú blokkokba sorolhatók: patológiablokk; súlyossági blokk; teljesítményblokk; leíró blokk; predikcióblokk; terápiablokk.

## A patológiablokk

Ide azok a feltételek tartoznak, amelyek a betegség tényének (vagyis a mintailleszkedés mértékének) megállapításában játszanak szerepet. A patológiablokk kialakításánál három alapvető szakmai (orvosi) szempontot kellett figyelembe venni:

— Egy tünet vagy tünetcsoport általában nem ugyanolyan mértékben erősíti a diagnózist, mint amilyen mértékben ugyanazon tünetek hiánya kérdésessé teszi az adott betegség megállapítását. Ennek megfelelően a patológiablokkban szereplő feltételekhez közelítő-távolító súlypárokat kellett rendelni. A közelítő súlyok az adott minta és a vizsgálati eredményhalmaz közelségének kiszámításában, míg a távolító súlyok a két halmaz messzeségének meghatározásában vesznek részt.

— Bizonyos esetekben egy tünetet többféle vizsgálattal (például többféle műszerrel: röntgen, ultrahang, CT stb.) ki lehet mutatni. Máskor ugyanazon tünetnek több megjelenési formája lehet. Nyilvánvaló, hogy ilyenkor a mintának tartalmaznia kell a jelenséget leíró összes feltételt, de több feltétel teljesülése nem jelenthet jobb illeszkedést, vagyis biztosabb diagnózist. (Például, ha ugyanazon agysérülést ultrahanggal és CT-vel is kimutatták.) Máskor viszont bizonyos tünetek együttes jelentkezése lehet csak az adott diagnózist megerősítő tényező. Tünetegyüttesek esetén az egymást helyettesítő vagy együttesen jelentkező feltételeket önállóan kiértékelendő feltételcsoportokba kell sorolni. A feltételcsoport feltételeihez nincsenek hozzárendelve súlyok, viszont a feltételcsoport az elemi feltételekhez hasonlóan önálló súlypárokkal vesz részt a mintaillesztésben.

— Egy betegségnek fontos komplex tünete lehet egy másik betegség, szimptóma diagnosztizálása. Ez azt jelenti, hogy egy mintában feltételként megfogalmazva hivatkozni lehet egy másik minta kiértékelésére. A hivatkozott minta illeszkedésétől függően a hivatkozás az elemi feltételekhez hasonlóan közelítő és távolító súlyokkal vesz részt az eredeti (hivatkozó) minta illeszkedésének megállapításában.

## A súlyosságblokk

Az ebben szereplő feltételek az adott betegség megállapítása esetén, vagyis csak akkor jutnak szerephez, ha a minta a patológiablokk kiértékelése alapján illeszkedik a beteg vizsgálati eredményhalmazára. A regisztrált betegség súlyosságát a diagnosztikai program a súlyosságblokk feltételeihez rendelt súlyossági faktorok segítségével határozza meg. A patológiablokkhoz hasonlóan a súlyosságblokkban is lehet konjunktív és diszjunktív feltételcsoportokra hivatkozni.

## A teljesítményblokk

Ez azokat a vizsgálatokat tartalmazza, amelyekkel a beteg gyermek azon teljesítményeit (például beszéd, járás, kommunikáció stb.) lehet mérni, amelyekben az adott minta által reprezentált betegség miatt korosztályához képest elmarad(hat). A teljesítményelmaradás mértéke azt mutatja meg, hogy a beteg a mintához tartozó teljesítményvizsgálatok alapján hány hónappal fejletlenebb vele egyidős egészséges társainál. A teljesítményekben mutatott elmaradás mértéke önmagában is fontos információ, ugyanakkor a betegség súlyosságát is kifejezi. Vagyis a betegség súlyosságának megállapításában a súlyosságblokkon kívül a teljesítményblokk is részt vesz.

## A leíróblokk

Azok a vizsgálatok tartoznak ide, amelyek meghatározzák ugyan az adott betegség jellegét (például: milyen típusú hydrocephalusról van szó), de sem a betegség tényének, sem súlyosságának megállapításában nem vesznek részt.

## A predikcióblokk

Ebben találhatók meg azok a vizsgálati eredmények (gyakran anamnesztikus adatok), amelyek valószínűsíthetik az adott betegséget. Ide tartoznak például az olyan rizikófaktorok, mint a koraszülés, terhesség alatti dohányzás és alkoholfogyasztás, amelyek gyakran jól körülhatárolható és statisztikailag is valószínűsíthető idegrendszeri sérüléseket okozhatnak az újszülöttnél. Természetesen a predikciós blokk sem vesz részt a betegség tényének és súlyosságának megállapításában, hiszen egy adott betegség diagnózisa nem lesz biztosabb csupán azáltal, hogy ismerjük kialakulásának feltételezhető okait. A predikciós blokkal az orvosi intuíció megragadása a cél, hiszen néhány adat felvétele után esetleg éppen ezek a nem diagnosztikus információk keltik fel az orvos gyanúját, és ezáltal megszabhatják a további vizsgálódások irányát.

## A terápiablokk

Mindazokat a gyógyszeres és egyéb kezeléseket, tréningeket tartalmazza, amelyeket az adott betegség diagnosztizálása esetén a gyógyítás érdekében alkalmazni kell. Itt lehetőség van offenzív és defenzív terápiacsomagok kialakítására, amelyek közül az orvos az eset jellegének, saját szakmai habitusának és felelősségének megfelelően választhat.

**Somogyi Péter**

## A Microsoft Test for Windows

# Test test ellen?

Egy valódi piacgazdaságban egy újonnan megjelenő termék fejlesztési költségének jelentős hányadát teszi ki az új produktum minden apró részletre kiterjedő, előzetes minőség-ellenőrzése.
Különösen igaz ez az egyik legjobban áruba bocsátható szellemi termékre, a szoftverre.
Míg külföldön egy szoftver teljes fejlesztési idejéből 70-80 százalékot, összes költségeiből 40-50 százalékot fordítanak az alapos tesztelési fázisra, addig például hazánkban...

Tisztelet a kivételnek! A szerző negatív tapasztalatai főként az elmúlt évek itthoni ügyviteli dBase/Cobol/Clipper/FoxBase-lázán alapszanak. A ráfordításokról szólva a külhoni gyakorlattal szemben nálunk zömében pont annak a fordítottja igaz, vagy még az sem. Gyakorta az élesben dolgozó alkalmazás sorsüldözött felhasználói tes(z)t(j)én a finisben szűrték ki az utolsó jó pár futásközbeni hibát.

Komoly, nagy szoftverházak is követnek el azért ilyen hibákat, miszerint a terméküket elegendő és körültekintő teszt nélkül értékesítik. Nem kell messzire menni, elég csak az MS-DOS hibás 3.2 és 4.0 változatára visszagondolni, vagy az MSC 5.0 és 6.0 következménye is: egy 5.1 és a 6.1. A javítás volt a kényszerű vezeklés eredménye. A Windows 3.0 1990 májusában még csak külsejében hasonlított az ugyanazon év októberében kiadott (szintén 3.0-ra hallgató!) verzióra. Néhány hete használjuk a Windows 3.1-est, és tapasztalhattuk, hogy ez sem makulátlan.

No, az előbb felsorolt programok kifejlesztője készítette a Microsoft Test for Windows programot (továbbiakban MTW), amelyet éppen a fenti kellemetlenségek megelőzésére találtak ki. Lássuk, milyen fegyvertárat ajánl a minőségért folytatott gyötrelmes, kényszerű, de nemes küzdelemhez a PC-s szoftvervilág első számú óriása!

(A programok, szoftverek, szoftverrendszerek „próbaüzemeléseinek", elkészítésük befejező fázisainak módszertani ajánlásait a Fogódzó rovatnak a témához kapcsolódó cikke foglalja össze.)

### Automatikus teszt

Az MTW nem más, mint egy, a szoftvertesztelést automatizáló eszköz Microsoft Windows 3.x grafikus környezetben fejlesztők számára. Windowsos alkalmazások minden részletre kiterjedő, felügyelet nélküli tesztelését teszi lehetővé. Az MTW kiaknázza az automatizmus előnyös tulajdonságait:

— Konzisztencia. Hajszálra ugyanazt a tesztsorozatot ismételhetjük meg, amennyiszer csak akarjuk.

— Pontosság. A mérési eredményeket egyáltalán nem befolyásolja a fáradékony tesztszemélyzet koncentrálóképessége.

— Alaposság. Az eredményeket naplózza, így azokat utólag kimerítően elemezhetik és összevethetik a személyzet tagjai.

Az MTW-vel tüzetesen vizsgálhatunk komplex alkalmazásokat anélkül, hogy az a megszokott munkarendet felborítaná. Szinte az összes olyan Microsoftos fejlesztő csapat jelentős időt, fejlesztési költséget (vagy — az angol nyelvterületen kívül — adaptációs, honosítási költséget) takarított meg, amelyik MTW-re támaszkodott:

— Kezelői felületek tesztjénél (olyan termékeknél, mint a Microsoft Excel vagy a Microsoft Word for Windows 2.0).

— Nem user-interface jellegű teszteknél (DLL = Dynamic Link Library-k és makrónyelvek).

— Batch jellegű teszteknél: éjszakai vagy hétvégi tesztelő futtatások emberi felügyelet nélkül.

— Helprendszerek ellenőrzésekor.

Az MTW-vel bármilyen Windows-alkalmazás tesztelhető — függetlenül attól, hogy milyen fejlesztő eszközzel (és programozási nyelven) készítették. Minden tesztszekvenciát egy Script nyelven írt kisebb programocska határoz meg, amely egy egyszerű utasításkészlet útján mondja meg a Test Drivernek, hogy mit és hogyan csináljon. A scriptek szimulálhatják a billentyűzetet és az egérelmozdulásokat, és természetesen a program ezekre adott válaszreakcióit. Ugyanazokat a felhasználói aktivitásokat (billentyűzéseket és egérmozgatást) teszteltethetjük többször is, és ez azért nagyszerű, mert ezeket kézzel bajos lenne akár csak kétszer is egyformán (azonos módon és sebességgel) végrehajtani.

Nagyon könnyű beletanulni a tesztelő scriptek írásába. Persze sok egyszerű scriptet meg sem kell írni, hiszen lényegesen könnyebb az emberi tesztelő működését (gombnyomásait és egéraktivitását) automatikusan feljegyezni, és később ezt a tárolt szekvenciát visszajátszani vagy editálni. A Script nyelv nagyon hasonlít a Visual Basic-hez. Mivel még mindig a Basic az egyik legáltalánosabban ismert programozási nyelv, így a fejlesztők és a programozók valószínűleg nem fogják nehéznek találni az MTW nyelvét.

A gyors elsajátíthatóság fontos szempont: ennek érdekében az MTW helyzetérzékeny helprendszert tartalmaz. Komplett tesztforgatókönyv hozható létre, amely magában foglalja a hibakezelést, az UAE-k (Unrecoverable Application Error) okának megkeresését, az időzítéseket (time-outs), és a tesztszekvencia futási sebességét egyaránt. Ha valaki nem ismerné a Basic-et vagy, csak a beállítottsága miatt idegenkedik tőle, még mindig írhat scriptet a FastTest segítségével. A FastTest a teljes MTW egy szűkített változata, amely angol köznyelvi kifejezésekkel dolgozik, és a legtöbb belső függvényt alapértelmezésben, emberi paraméterezés nélkül használja.

Az MTW nem igényel speciális hardvert, egy másik számítógépet, vagy egyedi töréspontokat a tesztelendő alkalmazásban. A megfelelő előkészítés viszont talán a legfontosabb tényező az

automatikus tesztelés alkalmazása esetén. (Erről is részletesen szól a „Tesztkontroll” című írás az 50. oldalon. — A szerk.) És addig kell faragni egy alkalmazást, míg minden egyes — az alkalmazásban szóba jöhető — konfigurációban ugyanazzal az MTW-tesztscripttel ugyanazokat az eredményeket nem kapjuk!

Abból a célból, hogy az MTW minimalizálja a Windows-erőforrások (resources) felhasználását és maximális rugalmasságot biztosítson, több kisebb-nagyobb DLL-ből tevődik össze. Ezek bármelyik olyan programozási nyelvvel együttműködnek, amelyek támogatják a DLL-ek használatát (Borland C++, MSC 6.x és 7.0, TopSpeed nyelvek, Zortech C++, Watcom C, Turbo Pascal for Windows, Visual BASIC, MS FORTRAN 5.1 stb.).

## Az MTW részei

— Test Driver (TestDrvr.EXE, MSTEST.INC és FASTTEST.INC), amely egy Basic interpreter, kiszolgáló környezetet ad tesztelő scriptek írásához és futtatásához. A Test Driver egy Windows alapú Basic nyelvjárás, olyan strukturált Basic-tulajdonságokat támogat, mint például a SUB és FUNCTION utasítás, vagy a tradicionális vezérlőszerkezetek, mint: IF ... THEN, WHILE ... WEND, FOR ... NEXT és SELECT CASE. IFDEF szimbólumokat is engedélyez a feltételes fordítás megvalósításának érdekében. Néhány jellemző szolgáltatása: előre megjósolhatatlan események elkapása (a felhasználódühítő UAE — Unrecoverable Application Error — okának felderítése), egyik tesztelő script meghívhat feltételrendszertől függően másikakat, a Test Driver saját ablaka minimalizálható vagy elrejthető, hogy ezzel elkerüljük a tesztelt alkalmazással az egymásra hatást, kis méret (maximum négyszer 64 kbájtos memóriaszegmenst foglal el, de a legtöbb életszerű szituációban ennél kevesebbet).

— Test Dialogs (TestDlgs.EXE és TestDlgs.DLL), segédprogram és függvénykönyvtár a programok vezérlésének, azok vizsgálatának vizuális összevetésére. Különösen alkalmas például ugyanazon program EGA és VGA üzemmódban való teszteredményeinek intelligens összevetésére. Az eltérő méretű és alakú, de ugyanazon funkciójú nyomógombok (buttons) között logikusan nem tesz különbséget ("fuzzy" logic, „hasonló, bár nem teljesen azonos” logika).

— Test Screen (TestScrn.EXE és TestScrn.DLL), képernyőtartalmak vizuális összehasonlításához. Működési elve: előre megadott koordinátájú helyekről „fényképeket” (screen-captures) készít, s ezeket hasonlítja össze tévedhetetlenül éles szemmel. Az eltérő képeken logikai XOR operátor segítségével emeli ki az eltérő részeket. Természetesen minden elképzelhető videokártyával képes együttműködni.

— Test Event (TestEvnt.DLL), olyan API-k (Application Program Interface — a program kezelői felülete), melyek billentyűzettel és egérrel kapcsolatos eseményeket szimulálnak. Maga a tesztelt alkalmazás nyilvánvalóan nem képes észlelni, hogy a vezérlőutasításokat nem emberi kéz által kapta. A 128-as ASCII kód feletti karaktereket is támogatja!

— Test Control (TestCtrl.DLL), függvénykönyvtár, mely információkat ad a programfolyam állapotáról, és az egyes rutinokat, vezérléseket név szerinti hivatkozással engedi manipulálni kívülről.

Példának okáért egy OK gomb mindenkori állapotát képes lekérdezni, de „le is tudja azt nyomni”, szemben a TestDlgs.DLL-beli szolgáltatással, amely csupán az OK gomb állapotát hasonlítja össze a dialog fájlban előzetesen tárolt állapottal.

Ára az Alaplap Posta útján 41 900 Ft + ÁFA.

**Herczeg József**

### Minimális hardver- és szoftverkövetelmények

— AT 286; 1 Mbájt RAM (2 Mbájt ajánlott); 3 Mbájt szabad harddiszk-kapacitás; vagy 1.2 Mbájt, 5.25", vagy 720 kbájt, 3.5" lemezmeghajtó;
— DOS 3.3 vagy magasabb verzió; MS Windows 3.0 vagy magasabb verzió;
— Egér javasolt.

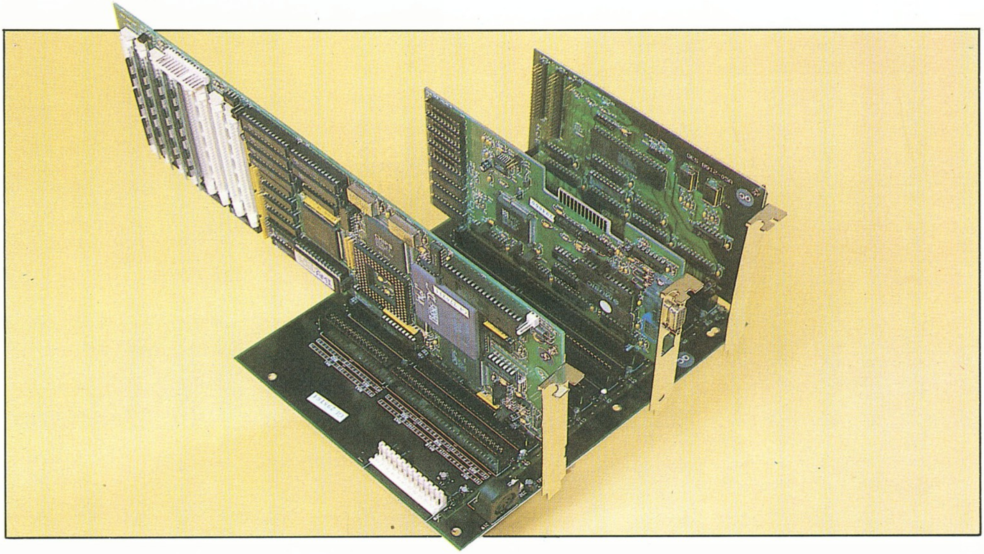
# Mindent egy kártyára!

## MULTIflex (MF Series) SYSTEM

Ez a sarok COMPFAIR '92 A pavilon, 110-es stand. 10% Önnek a COMPFAIR ideje alatt kedvezményt jelent.

**SZILICIUM ELEKTRONIKA Kft.**

BP. 1065 NAGYMEZŐ U. 66. · TEL.: 1321-912 · FAX: 1327-572



A MULTIflex rendszer új szemléletet és új eszköztárat ad számítógépünk korszerűségének megőrzéséhez. A szokásos alaplap helyén csak üres foglalat van, s egy könnyen cserélhető kártya hordozza a processzort, a RAM-ot, a BIOS-t és az akkumulátort. A többi szokásos kártyával az egységesített AMI BIOS teremti meg az együttműködést.

A CPU kártyára biztosított 2 év garancia önmagában is jelzi a rendszer megbízhatóságát és minőségét. A kártya japán technológiával készül, többféle felépítésben:

— AT 286, 16 MHz, 1 MB RAM
— AT 386SX, 20 MHz, 2 MB RAM, 128 K cache
— AT 386DX, 33 MHz, 4 MB RAM, 128 K cache
— AT 486SX, 20 MHz, 4 MB RAM, 128 K cache
— AT 486DX, 33 MHz, 4 MB RAM, 128 K cache
— AT 486DX, 2—50 MHz, 8 MB RAM, 256 K cache (DUAL CLOCK)

Számítógépünk megbízhatóságát szolgálja a teljes Fujitsu merevlemez-választék, a WD Paradise VGA kártya, a 6-féle monitor... és a működtetéshez szükséges összes tartozék.

***A dealereket és a viszonteladókat is várja a MULTIflex berendezések kizárólagos hazai forgalmazója:***

INFORMÁCIÓKÉRÉS: 07 ▶



## Őszi Comdex: 1992. november 16–20.

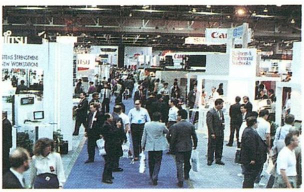
# Las Vegas csak egy ugrás!

Miről nevezetes Las Vegas?
Hát elsősorban játékkaszinóiról.
No meg arról, hogy a szórakoztatóipar valamennyi műfajában otthont ad a csúcsteljesítményeket felvonultató színpadi látványosságoknak.
A beavatottakon kívül azonban kevesen tudják, hogy Las Vegas a csúcstechnológia bemutatásában is a világelsők között foglal helyet.
Aki tehát még nem érzi magát eléggé beavatottnak, az olvassa el alábbi előzetesünket a Comdexről.

A vásárlátogatás a jó szakembereknek befektetés. Használval mindazok tisztában vannak, akik mohón lesik a technikai újdonságokat, akik szakmai kapcsolatokat akarnak kiépíteni és ápolni, akik ötleteket akarnak gyűjteni saját munkájukhoz... és természetesen mindazok, akik venni vagy eladni akarnak. Ilyesmit sehol máshol nem lehet olyan koncentráltan és olyan rövid idő alatt megvalósítani, mint a szakvásárokon és kiállításokon. Minél teljesebb és rangosabb egy ilyen bemutató, minél gazdagabb kiegészítő rendezvények kapcsolódnak hozzá, annál eredményesebben.

Anyagi korlátok miatt a magyar szakemberek többsége inkább csak az európai vásárokon bukkan fel. Egy tengerentúli rendezvényre ellátogatni megfizethetetlennek tűnik... és nagyon sokszor valóban az is. De vannak kivételek. Kolumbusz után pontosan 500 évvel fel kell fedeznünk Amerikát. Azon belül is Las Vegast és a Comdexet. Jóval olcsóbban is megtehetjük, mintsem gondolnánk!

Hogyan lett a Comdex a világ egyik legrangosabb számítástechnikai bemutatója és szakmai találkozóhelye? Hát nem egyik pillanatról a másikra, pedig azt feltételezhetnénk, hogy Kalifornia állam közvetlen szomszédságában és a „szilicium-völgytől" csak néhány száz kilométerre gyorsabban is berobbanhatott volna a szakmai közéletbe az 1979-ben 157 kiállítóval és 4 ezer látogatóval elkezdődött rendezvénysorozat. Az idei Comdex/Fall kiállító cégeinek száma azonban már várhatóan meghaladja a 2 ezret, a több helyszínen „elszórt"



bemutatók összes alapterülete a 200 ezer négyzetmétert és a látogatók száma a 130 ezret, köztük 20 ezer külföldivel. Az eseményről mintegy 1800 szakújságíró ad majd tudósítást a világ minden részére. Mindezt azonban 14 év alatt, egyenletes növekedéssel érték el.

A hannoveri CeBIT mind alapterületét, mind a látogatók számát tekintve felülmúlja ugyan a Comdexet, de Európában mindig is sokkal koncentráltabb volt az árubemutatók szervezése, erősebbek a hagyományai a nagy vásári seregszemléknek. (Amerikában a Comdex a legnagyobb szakvásár, megelőzve más ágazatok rendezvényeit is.) A Comdex nemzetközi tekintélyét nem is annyira mennyiségi, mint inkább minőségi mutatói adják, ami egyrészt annak tudható be, hogy sok új fejlesztési eredmény első nyilvános bemutatását, „világpremierjét" a Comdexre időzítik, másrészt igen tartalmasak a vásárt kiegészítő szakmai rendezvények, konferenciák. Idén például 5 szekcióban az alábbi témák köré csoportosulnak az előadások:

— Új eszközök szekciója: multimédia, képfeldolgozás.

— Illesztési szekció: nyílt rendszerek, hordozható gépek.

— Csatorna szekció: hálózatok, eszközgyártás.

— Gazdasági megoldások szekciója: vállalati stratégia, technológia.

— Nemzetközi szekció: Amerikán kívüli piac, együttműködési lehetőség.

Ami a kiállítókat illeti, elmondható, hogy ott van „mindenki, aki számít". És kik a Comdex legszorgalmasabb látogatói? Disztribútorok, viszonteladók, rendszerszervezők, hardver- és szoftverkészítők, vállalati döntéshozók és nagybani bevásárlók. A rendezvény üzleti szándékairól árulkodott már indulásakor a Comdex elnevezés is, amely a COMputer Distribution EXhibition rövidítése (számítógép-forgalmazási kiállítás). A vásárt szervező Interface Group 1981-ben lehetőséget látott arra, hogy az őszi Las Vegas-i Comdex mellett elindítson egy tavaszi „vándorkiállítást", amely szintén sikeresnek bizonyult és évente másutt kerül sorra, így 1992-ben a Comdex/Spring helyszíne Chicago volt, 1993-ban Atlanta lesz.

Saját szemünkkel látni valamit — sokkal többet jelent, mint csupán hallani és olvasni róla. A távolság miatt Amerikába rendszeresen „átugrani" talán ezentúl is kevesen tudnak majd, de a költségekben nincs akkora különbség, hogy néha — vagy csak „egyszer az életben" — akár cégünkkel, akár egyénileg ne engedhetnénk meg magunknak egy ilyen szakmai kirándulást. A számítástechnika legnagyobb európai rendezvényére, a CeBIT-re évente és tömegével zarándokolnak el a hazai szakemberek.

Az az út pedig, amit most az Alaplap szerkesztősége a Mercator Travel utazási irodával közösen szervezett az őszi Comdexre, nem kerül többe egy CeBIT-látogatás kétszeresénél. Lapzártakor már csak 1-2 kiadó helyünk volt, de a jövő évi Comdex rendezvényekre (májusban Atlanta, novemberben Las Vegas) nagyobb létszámú csoportokat szervezünk, és az akcióba elsősorban az Alaplap olvasóit szeretnénk bevonni. A részletekről következő számainkban írunk.

**Faklen Pál**

## Kódkönnyítés a kódoláshoz

# Minő menő menük!

A programozói munka egyik legutálatosabb része, amikor az egyes alkalmazások felhasználói menüit kell kódolnunk. Aki már próbálta, nem szívesen vállal újabb hasonló feladatot. Szerencsére a szoftverfejlesztők is mélyen átérzik ezt a problémát, így segédeszközök széles skáláját bocsátották rendelkezésünkre. A SolarSoft kínálatából sem hiányoznak.

Az adatbeviteli képernyők kialakítása csak a probléma egyik oldala, a másik a tevékenység hozzárendelése az egyes menüpontokhoz, úgy, hogy a mindennapi használat során előforduló tévesztési lehetőséget minimálisra csökkentsük. Tekintsük át először a SolarSoft kínálatában a menügenerátorok választékát (zárójelben a lemez sorszáma).

### Menügenerátorok

**Dosmenu** (#017) — Egyszerű menürendszer, Basic-ben készült. Az ezzel készített menüben 30 parancs helyezhető el. 15 batch program és 15 DOS parancs választható ki — ezek a felhasználó igényei szerint editálhatók. Ne vezessen félre senkit a program neve! Ez nem az a menügenerátor-típus, amiről a fáradt programozó álmodik! Ez a felhasználók (a minimálisan képzett felhasználók) életét teszi könnyebbé. Nem kell ugyanis hosszú, több paraméteres parancssorokat begépelniük, a kisebb batchprogramok egyszerűbben indíthatók.

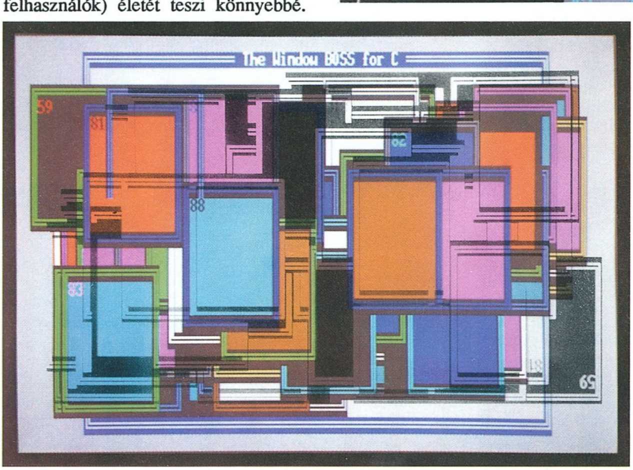
**Automenu** (#018) — Igen ügyes menügenerátor, a Dosmenuhöz hasonlóan batch programok indítására szolgál. Megakadályozhatja a felhasználókat az akaratlan kezelési hibák elkövetésében. Az egyes feladatok indítását egy szám begépelésével végezheti az, akinek a jelszóvédelem hozzáférést engedélyezett. Több menü/almenü fűzhető egymás után. (Részletes ismertetését az Alaplap 1991. májusi számában olvashatják.) Az így kialakított rendszer képernyőkímélő.

**Easy Access** (#234) — Menüző előtétrendszer, amelyből korlátlan számú egymásba ágyazott menü útján indíthatók a programok. Jelszavas, hierarchikus védelmet is kialakíthatunk. A program tartalmaz naptárt és emlékeztetőt, amely naponta, hetente, havonta, évente stb. figyelmeztethet bizonyos dolgokra.

A menügenerátorok mellett számos olyan segédeszközt találunk, amely a Basic, Pascal és C nyelven programozó kollégák számára megkönnyíti az adott programnyelven a menük létrehozását,




különös tekintettel az adatbeviteli képernyő kialakítására.




### Basic

**A Turbo Basic Tools** (#005) lemezen több más program mellett menüző és menüző+ellenőrző programot is találunk, mind forrásnyelven, mind pedig végrehajtható állapotban. Az első program képernyőmaszkokat állít elő, míg a második a begépelt input adatok helyességét is ellenőrzi.

A **QBSCR Screen Routins** gyűjtemény (#373, 3 lemez) szubrutinjai közül a MakeMenu%, SubMenu%, BuildScreen%, MultiMenu%, BuildScreen% alkalmazása biztosan gyorsítja majd munkánkat. A demóprogram igen látványosan mutatja be a csomag által nyújtott lehetőségeket. Ez a 3 lemez sokak szerint a SolarSoft programkönyvtár egyik legjobb darabja.

A **QBTree & QBUtils** (#486) segédprogram-gyűjteményben Glenn Miller Topmenu2-je érdekes a menükészítők számára. Ez egy remek menüprogram teljes forráskóddal (Quick Basic), plusz egy menüdemó, szintén forráskóddal együtt.

A szerző saját bevallása szerint elkeseredésében fejlesztette ki, ugyanis hiába keresett olyan segédprogramokat, amelyeket Quick Basic programrendszerében felhasználhatna a menükészítéshez.

Minden esetben terjedelmes könyvtárakat kellett volna nagyrészt feleslegesen beszerkesztenie azért, hogy néhány rutint használhasson. A Topmenu 11 K-nál kevesebb helyet foglal el, ha beillesztjük saját programunkba.

## C nyelv

A **Window-Boss** (#204, 2 lemezen) neve magáért beszél. Teljesen professzionális ablaktechnika-kezelő C könyvtár. Garantáltan gyors rutinokat, mintaprogramokat tartalmaz, amelyekkel redőny-, pop-up menük, ablakok mozgatása, on-line help-rendszer készítése oldható meg MSC 5.x, Lattice C, egyéb C fordítók alatt. (Teljes támogatás Borland Turbo C-hez.)

A **C-Window Toolkit** (#334, 2 lemez) ablakkezelő bővítéseket tartalmaz: ablaktechnika, pop-up, pull-down menük, segédablakok, egységesített beolvasási lehetőség. Sok C és H mintapélda, jól használható MAK és PRJ állományok (a gyors fordításhoz) könnyítik a fejlesztők munkáját.

A **C++ Toolkits #2** nevű lemez (#499) OOP kiegészítéseket tartalmaz. Ezek között Boss néven érdekes megjegyzéseket találunk a Window Boss használatához. Ezzel a Turbo C++ ANSI fordítóra érvényesek.

## Pascal

A **Turbo Designer** (#270) Turbo Pascal 4.0 és 5.0 menü- és helptervező segédprogram forráskódgenerátorral. Nagyon tetszetős, könnyen használható, hatékony program, nagyon sok munkától kíméli meg a programozót. A SolarSoft Programkönyvtár egyik legnépszerűbb darabja.

A **TOP — Turbo Overdrive Package** (#312) segédrutinjai közül a Menu123 Lotus-szerű menük, a Menubox pop-up menük, a Menupull pull-down menük kialakítására szolgál. A mellékelt TPU-k forrásprogramjai is megtalálhatók a lemezen!

**Szőke Péter**

# Sétálunk... az aknamezőn

Volt Ön katona? „Járt" már aknamezőn?
Ha még nem, itt a teljesen veszélytelen alkalom, hogy kipróbálja ezt az érzést!
A SolarSoft Programkönyvtár 522-es lemezén található Mine Field (magyar jelentése: aknamező) program lenyűgöző logikai-deduktív játék.
Valószínűleg már sokan ismerik a Windows alatt futó változatot.
A többieknek pedig íme, itt van a DOS-os verzió.
(Sajnálatos, hogy csak Microsoft egérrel használható.)

A játék lényege, hogy a rendkívül változatos tizenöt pálya mindegyikén megtaláljuk, illetve a jobb oldali egérbillentyűvel kijelöljük az elrejtett aknákat. A pályák alaprajza az egyszerű téglalaptól a várkastély formán keresztül a Batman-logóig terjed. A pályák erőssége a megkeresendő aknák számától függ, de ezt magunk is szabályozhatjuk. Vigyázzunk! Ha nem jó a kijelölésünk, a játék végén minden akna felrobban.

Kezdéskor van a legnehezebb dolgunk. Hasraütéssel kell „tisztának" minősítenünk az egyes mezőket. Ha azonban szerencsénk van, akár a fél pálya is feltárulkozik előttünk. A részben láthatóvá vált pálya további „tisztogatásakor" vegyük figyelembe a színes kockákba írt számokat! Ezek a detektorok, amelyek segítenek eldönteni, hogy hol található az akna, ugyanis a számok azt jelölik, hogy 1, 2, 3 vagy több mező alatt lesz-e akna a számozott kockával érintkező nyolc közül. Általános szabály, hogy ha valamelyik 1-aknás jelölésű mező mellett biztosan megállapítottuk a telepítés helyét, a másik 7 kockán már nem lehet akna. Akkor sem, ha első pillantásra netán másként érezzük. Mielőtt az „aknaszedést" befejeznénk, nézzük át a pályát még egyszer! Ilyenkor még van lehetőségünk arra, hogy kijelöléseinket felülbíráljuk. Vagyis ha úgy ítéljük meg, akkor a jobb oldali billentyű újbóli leütésével eltörölhetjük és egy másik kockára tűzhetjük ki az akna helyét jelölő piros zászlót.

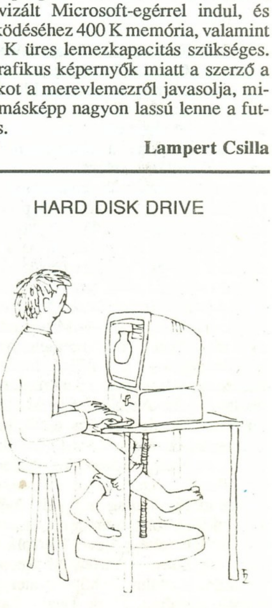
A bejelentkező setup-képernyőn választható ki a pálya, szabályozható a hang ki/be kapcsolása, illetve itt változtathatjuk meg az alapértelemezésben telepített aknák számát egy-egy pályán. Minimálisan 10 akna telepíthető — a maximum megegyezik a pálya mezőinek számával.

A programot Charles A. Hardy írta, felhasználva hozzá a Turbo Pascal 6.0, az Expert Software Tools, a GX Development Series és a TechnoJock's Turbo Toolkit szoftvereket.

A program csak VGA monitoron, aktivizált Microsoft-egérrel indul, és működéséhez 400 K memória, valamint 350 K üres lemezkapacitás szükséges. A grafikus képernyők miatt a szerző a játékot a merevlemezről javasolja, mivel másképp nagyon lassú lenne a futtatás.

**Lampert Csilla**

HARD DISK DRIVE



## 140 program

# Windows-újdonságok — kimazsolázva

Egyre nő a Windows-rajongók tábora. Reméljük, mindannyian megelégedéssel nyugtázzák, hogy a SolarSoft eddigi viszonylag szegényes Windows-kínálata ugrásszerűen bővült. Öt lemezen (#544 — #548) Windows környezetben játszható játékokat találnak (36 darabot), míg az #549 — #566 lemezeken 104 (!) alkalmazást. Nincs helyünk ilyen sok program részletes ismertetésére, ezért ad hoc jelleggel mutatunk csak be egy-egy érdekességet. Nem feltétlenül a leghasznosabbakat, hanem azokat, amelyekben felfedeztünk valami egyénit, humorosat vagy jól sikerült megvalósítást.

ANTS (#556) — A nevéhez híven hangyákkal hinti tele a képernyőt. Jönnek-mennek, tülekednek a kenyérmorzsáért, elbóklásznak a megnyitott ablakok, ikonok mögé, elszaladnak egymástól, visszajönnek. Kellemes látvány, a szerző tulajdonképpen animációs hátteret akart készíteni a Windows-hoz.

COOKIE (#549) — Tulajdonképpen a Fortune Cookie rövidítése. Tündériek a szólások, magát az adatállományt végigolvasni is élvezet. Összeállítása sem lehetett akármilyen munka, hiszen kinyomtatva 9 sűrűn teleírt oldal. Ezek az angol szólások, Murphy-törvényszerűségek lecserélhetők, saját magyar sziporkákkal helyettesíthetők. A program installálásakor megadhatjuk, hogy milyen időközönként villanjon be a képernyőre egy-egy véletlenszerűen kiválasztott idézet. Lehet, hogy kicsit infantilis, de azért kedves.

QUOTE11 (#552) — Minden napra egy idézetet ad. Ezek megjelenhetnek véletlenszerűen, de rákereshetünk a szerző nevére, egyes szavakra, szókapcsolatokra is. Az előző verzió elsöprő sikerén felbuzdulva fejlesztették tovább a programot. Magukat az alkotókat is meglepte a nagy érdeklődés. A Cookie-val ellentétben, amely „csak" humoros, itt az idézetek magvasabbak. Néhány név, akinek több mondása is szerepel: Albert Schweitzer, Lewis Carroll, Peter Ustinov, Jimmy Carter, Shakespeare, Picasso, Buddha, Schopenhauer, Jane Austin, Kipling, Mark Twain, Samuel Goldwin, Golda Meir, Voltaire, Arisztotelész ... A felsorolás magáért beszél. Természetesen itt is variálhatunk. Az adatállományt azonban csak Windows-editorral módosíthatjuk.

CUBIC (#558) — Háromdimenziós kockajáték. A bűvös kockát kell kiraknunk. Billentyűzetről is vezérelhető, ha elfáradtunk vagy meguntuk, automata önkirakásra is átállíthatjuk.

GRAVITY (#559) — Égi szimulátor. A SkyGlobe sikere alapján feltételezhető, hogy ez is felkelti sokak érdeklődését. Nem akármilyen szimulátort tartunk kezünkben! A Naprendszer működésébe avatkozhatunk be. Választ kapunk számtalan „mi lenne, ha..." kérdésre a bolygómozgással kapcsolatban. A kezdő képernyőn középen a Nap, körülötte a Föld, a Merkúr, a Vénusz, a Mars és a Jupiter. Mind az öt bolygónál egy-egy vektor jelzi a pillanatnyi sebesség irányát és nagyságát. A szimulációs menüpontból elindítva mozgásba jön ez az állókép. Most jön az érdekesség: megállíthatjuk, a sebességi vektorokat tetszőlegesen módosíthatjuk (a Föld például foroghat visszafelé), majd pedig az így beállított sebességekkel a tömegvonzás törvényei alapján újraindulnak a bolygók.

ICNMAGIC (#560) — Ikoneditor Microsoft Windows-hoz. Professzionális kivitelű saját (16-színű) ikonokat készíthetünk, szabványos (*.ICO) ikonállományokat editálhatunk, módosíthatjuk a végrehajtható (*.EXE) állományokból vagy a képernyőről átvett ikonokat. Rajzolhatunk, vonalakat húzhatunk (1—8 pixel vastagságban) két pont között, kitölthetünk mintákat a választott színnel, a rajzterületre ikon méretű képeket hozhatunk be a képernyőről a snapshot funkcióval, ellipsziseket, köröket, téglalapokat rajzolhatunk. A Clear választásakor az ikon üres részein áttetszik a tetszőleges háttérszín, míg Inverse esetén a háttérszín ellenpárja jelenik meg.

PLIFE (#563) — Közismert cellajáték. A keletkezés-elhalás folyamatát mutatja be: háromnál több szomszéddal rendelkező üres cella életre kel, kettőnél kevesebb vagy háromnál több szomszéd esetén az élő cella elhal. Turbo Pascal for Windowsban íródott a program, az ObjectWindows felhasználásával. Forráslistát is mellékeltek a programhoz.

PULSAR (#563) — Postscript/PCL átkapcsoló. Kifejlesztését a szerzők munkahelyi körülményei inspirálták. Megunták ugyanis, hogy Windows környezetben állandóan át kellett kapcsolni a HP LaserJet nyomtató PostScript és PCL módja között. Ebből azután egy általánosabb megoldás született. A PULSAR ugyanis magában foglalja a HP LaserJet család leggyakoribb nyomtatási funkcióit, továbá néhány Windows funkciót (Control Panel, Print Manager, Printer Setup). Visual BASIC-ben írták, szükség van a VBRUN100.DLL run-time modul jelenlétére is. (Ez nemcsak ennél a programnál fontos, ezért hívjuk fel a figyelmet arra, hogy a modul a SolarSoft #547 számú lemezén található, három játékkal együtt.)

WINPLAY (#566) — Zene, Windows forrással, 6 mintadallammal.

WBAR18 (#565) — Vonalkódot generál Windows környezetben, bemásolható a vonalkód a Clipboardba, innen pedig kivágható és beilleszthető más programokba (Windows Write, Microsoft Word for Windows, Paintbrush, egyéb olyan Windows programok, amelyek képesek bitmap és metafájl fogadására). A WBAR18 a következő vonalkódtípusokat ismeri: 9-ből 3, UPC-A, Interleaved 5-ből 2, EAN13, EAN8, Codabar.

**Verebély Pálné**

## Játékok az Ablakban

# Logika és lövöldözés

A SolarSoft újdonságai között 5 lemezen összesen 36 olyan játékprogramot találunk, amelyek Windows grafikus felhasználói felületen futtathatók. Ezek közül mutatunk be most hármat.

Mindhárom logikai játék VGA monitort igényel, és nem árt, ha van hozzá egerünk, bár ez nem feltétlenül szükséges. A Bang!Bang! a Solarsoft programkönyvtár 544-es, a Winpuzz az 548-as, a Symbol-Rechen pedig az 546-os lemezén található.

### Bang!Bang!

Az első játék neve Bang!Bang!. Ez is logikai játék, bár első látásra egy kicsit félelmetes és vérszomjas. Két ágyú lövöldöz egymásra addig, amíg el nem találja valamelyik a másikat. Az egészet az teszi érdekessé, hogy a két játékos a lövést többféleképpen befolyásolhatja: változtathatja a lövés szögét és erősségét, s hogy a dolog nehezebb legyen, a két ágyú között szintkülönbség is lehet, de még azt is beállíthatjuk, hogy fújjon-e a szél. Lőni pedig pontosan kell, mert a játékot csak úgy nyerhetjük meg, ha az ágyúgolyó telibe talál.

A játékot egy vagy két személy is játszhatja, egy ember esetén a gép nem lő, hiszen a számítógép nálunk sokkal gyorsabban oldja meg a differenciálegyenletet. (Nem elírás: tényleg differenciálegyenlet megoldása áll a háttérben.) A találat pontosságát be lehet állítani, de ki/be lehet kapcsolni a szelet, és azt is, hogy a golyó lyukat üssön-e a hegybe, vagy ne. A játék egyszerűsége ellenére (vagy talán éppen azért?) rendkívül élvezetes.

### Winpuzzle

A második játékot Shenk & Horn Winpuzzle-nak hívják, az EXE fájl neve Winpuzz. Ez egy klasszikus 15-ös játék, gyönyörű képekkel. Háromféle ábra közül választhatunk, az egyik egy óra, a másik egy ház, a harmadik pedig egy jelenet a Frédi és Béni rajzfilmből. Mindegyiket kétféle felbontásban játszhatjuk, lehet 3 x 3-as vagy 4 x 4-es a tábla. Segítségként, ha akarjuk, láthatjuk a kirakandó ábrát a játéktábla mellett.

### Symbol-Rechen

A harmadik játék neve Symbol-Rechen. Ez a Füles című lapban is gyakran közölt logikai feladattípus számítógépes feldolgozása.

A legegyszerűbb, ha egy konkrét példán mutatjuk be.

```
ABCD  +  EFBC  =  GHIC
 -        -        -
ABAH  +  IIAA  =  BHFI
 =        =        =
OOFC  +  OGEI  =  JDJA
```

Oldjuk meg együtt! A J betű 1-et ér, hiszen az alsó vízszintes sorban egy két- és egy háromjegyű számot adunk össze. Emiatt G kilenccel egyenlő. A bal felső szám utolsó jegye 0, ez az első sorból látszik. A középső oszlop számainak első jegyei mutatják, hogy E eggyel nagyobb I-nél, a következő jegyekből pedig az következik, hogy I eggyel nagyobb F-nél. Tehát E kettővel több, mint F. Az alsó sorban F + E (+1, ha van maradék) = 1 vagy 11, ami csak úgy lehet, ha F négy, E pedig hat. Eszerint I egyenlő öttel. Félúton a megoldás felé a táblázat így fest:

```
ABCO  +  64BC  =  9H5C
 -        -        -
ABAH  +  55AA  =  BH45
 =        =        =
004C  +  09EI  =  101A
```

A B betű nyolcat ér, ez az utolsó oszlop első jegyeiből derül ki (nincs átvitel az előző oszlopból). A középső oszlop szerint most A egy vagy kettő lehet, de az egyes már foglalt, így az A kettő. Az oszlop utolsó jegyei alapján C = 7. Már csak a H hiányzik és ezt az első oszlop utolsó jegyeiből tudjuk kiszámolni, így ez 3.

Az eredmény:

```
2870  +  6487  =  9357
 -        -        -
2823  +  5522  =  8345
 =        =        =
0047  +  0965  =  1012
```

Mindez így elég bonyolultnak látszik, de a játékban sokkal egyszerűbb.

A képernyő bal oldalán látható a feladvány, a számok helyett színes ábrákkal. A jobb oldalon egymás alatt vannak ezek az ábrák és mellettük a még lehetséges számok. Így a játék áttekinthetővé válik. Az irányítás akkor a legegyszerűbb, ha van egerünk. Ekkor, ha a jobb oldali táblázatban rámegyünk a megfelelő számra, és lenyomjuk a jobb gombot, az lesz az ábra értéke, és azt a bal oldalon rögtön át is írja. Ha a bal gombot nyomjuk meg egy számon állva, azt kizárjuk a lehetőségek közül. A bal alsó sarokban négy gombot láthatunk német felirattal:

| | |
|---|---|
| Neue Aufgabe | Új feladvány |
| Pause/Weiter | Lemerevítés/Tovább |
| Noch einmal | Még egyszer ugyanaz |
| Ende | Vége |

Ezek a funkciók elérhetők a képernyő tetején lévő legördülő menü Spiel részén is. Itt a Tip-ben segítséget kérhetünk, az Auflösung-gal teljes megoldást kaphatunk, az Optionen pontban másféle ábrákat, egérgombcserét kérhetünk, vagy azt, hogy ne mutassa az időt. Itt lehet a három szint közül is választani. Az utolsó felirat, a Highscore, a szintenkénti leggyorsabb eredményeket tárolja. Ha rossz számot írunk be az egyik ábrához, azt a Tip pont utolsó opciójában javíthatjuk. Ez a játék nagyon jó kikapcsolódás, kellemes szórakozás.

**Gerlits Judit**

# Őszi SolarSoft újdonságok

**Chemie-Star** — Segédeszköz kémiatanuláshoz. A periódusos rendszer 107 elemének 7-7 legfontosabb jellemzőjét jeleníti meg, és további 26 jellemző is választható. Kémiai számítások oktatása, kikérdezése, ellenőrzése.

**Q&A Study Aid** — Az elterjedt kereskedelmi szoftver shareware változata. Tetszőleges szakterülethez generálhatunk vele tesztrendszert. 40.000 kérdést/választ tartalmazhat az adatbázis. A tesztadatok ASCII állományban vannak és egyszerűen bevihetők, módosíthatók. A tesztállomány feltöltése után a „vizsgáztatás" eredményei megjeleníthetők, tárolhatók. Az állományok jelszóvédelemmel titkosíthatók.

**Title Maker** — GIF konverter. A grafikát önmegjelenítő .EXE programmá alakítja át. Mindig négy képet ment el: Hercules, CGA, EGA és VGA formátumban. Így a képek gyakorlatilag bármilyen grafikus rendszerrel megjeleníthetők. Igen praktikus demók, logók használatakor.

**ProWindows Lite** — Jól sikerült virtuális ablakkezelés egérrel. Itt a virtuális azt jelenti, hogy az ablak mérete nagyobb lehet a képernyőnél. Az ablakot eltolhatjuk, elrejthetjük, méretét megváltoztathatjuk, görgethetünk.

**Wolfenstein 3-D** — Az egyik leggyorsabb, legszebb VGA játék, 2 lemezen. A SoundBlaster hanglehetőségeit maximálisan kihasználja.

A történet is érdekes: a játékos az a „rossz fiú", akinek be kell jutnia a náci Wolfenstein kastélyba, és meg kell akadályoznia a „Vasököl" hadműveletet (ez a verhetetlen hadsereg felállítását célozza meg). Az ördögi orvosprofesszor ugyanis a holttestekből akarja a verhetetlen hadsereget kialakítani.

**Verebély Pálné**

## SolarSoft a hónap témájához

| | |
|---|---|
| 043 Droege | NYÁK-tervező |
| 044 Cascade-2D | NYÁK-tervező (egér) |
| 079 PC-Calib | Műszerkalibrálás |
| 090 Electron | Rádióamatőröknek |
| 093 Differential Equation | Deriválás, integrál... |
| 121 XYSee | Függvényábrázolás |
| 122 XYPlot | Grafikus egyenletmegoldás |
| 136 Vibrating | 3- és 4-dimenziós rajz |
| 136 Vibrating | 3- és 4-dimenziós rajz |
| 160 Vanilla | Mesterséges intelligencia |
| 176 CC-Surveyor | Földmérőknek |
| 257 Crystal | Molekula/kristály modell (EGA) |
| 324 Draft Choice | Mérnököknek (EGA/CGA/Hercules) |
| 325 Digitized Voice | Fotózható NYÁK-tervvel (AT) |
| 326 Speech Digitizer | Kapcsolási rajzzal (XT-n is) |
| 375 Personal APT | NC programozás mérnököknek |
| M02 Simon | Matematikai problémák, szimuláció |
| M05 Algebra | Numerikus algebra, mátrixok |

# Hangsztár zenés napokra

Augusztusi számunkban megígértük, hogy a zenélő SolarSoft-újdonságok harmadik darabját is bemutatjuk olvasóinknak. Az Audiostar mind SoundBlaster (VOC), mind pedig VoiceMaker II (ADS) hanganyag feldolgozására alkalmas. A jövőben AdLib és VoiceMaker I (SND) irányba kívánják továbbfejleszteni, így ezen mintákat és állományokat is kezelni fogja.

Az Audiostar szerzői Lars Schenk és Frank Horn, németek, így a teljes dokumentáció is német nyelvű. A könnyebb áttekinthetőséget szolgálja a grafikus környezet, amelybe az AudioStar digitalizált mintáit betölthetjük és feldogozhatjuk. A minták egy ablakban, görbeként jelennek meg. Ezeken könnyen észrevesszük a szüneteket, a hangosabb, halkabb részleteket, így nem szükséges hosszas fejtörés az esetleges módosításokhoz.

A rendszer szolgáltatásai:

— Hangminták feldolgozása.

— A SoundBlaster és VoiceMaker II támogatása (SoundBlaster használatakor ügyeljünk arra, hogy az Audiostar számára a CT-VOICE.DRV állomány elérhető legyen!).

— Grafikus környezet, redőnymenükkel.

— VGA, SVGA, MCGA, EGA és Hercules grafikus kártyák támogatása.

— A leggyakoribb műveletekhez „hotkey" billentyűkiosztás.

F2: A betöltött minta elmentése.

F3: Állománybetöltés.

F7: A munkamemóriában található dallam lejátszása.

F8: Blokktartalom lejátszása.

F9: A képernyőn látható rész lejátszása.

F10: Teljes minta lejátszása.

ALT-F2: Blokktárolás.

ALT-F3: Blokkbetöltés.

ALT-X: Kilépés a programból.

INS: Pufferből hozzáfűzés.

DEL: Blokktörlés.

TAB: Információs ablak megjelenítése.

— Mind egérrel, mind billentyűzetről vezérelhető.

— Különböző minták felvétele és lejátszása.

— Sokrétű blokkfeldolgozási lehetőség.

— Minták keverése.

— Különböző visszajátszási lehetőségek.

— Blokkok betöltése és tárolása.

— Visszhangosítás.

— Blokkok átfordítása.

— Hangerő-változtatás.

— Kivágás/beillesztés.

— Belső hangszóró vagy VoiceMaker II hangszóró használata.

— ADS-VOC és VOS-ADS átalakítás (így VOC állományok is meghallgathatók a PC belső hangszórójából).

— Audiomaster II (Amiga IFF) állományok is betölthetők.

**V. M.**

Ne feledkezzünk meg régi kedvencünkről, a Pianomanről se!
Három új SolarSoft lemezen találunk hozzá dallamokat:

#569 — Barokk művek
#570 — Beethoven
#571 — Karácsonyi koncert

# Irodatechnika

azoknak, akik eddig még csak a kirakatokban láthatták.

*Nyugati, használt, felújított berendezések:*

| | | | |
|---|---|---|---|
| ■ Fénymásolók | 40 000,– | ■ IBM XT számítógépek | 16 700,– |
| ■ Monokróm monitorok | 2 500,– | ■ Nyomtatók | 6 000,– |
| ■ VGA-monitorok | 15 000,– | ■ Telefonok | 450,– |
| ■ Commodore monitorok | 2 500,– | | |

+ÁFA-s ártól

***SZINT*** 1118 Budapest XI., Zólyomi u. 6/B
Telefon: 185-1337, 186-1278 Fax: 186-9220

*Keressen bennünket a COMPFAIR kiállításon is!*

INFORMÁCIÓKÉRÉS: 16 ▲

# Ne importáljon!

**Mágnescsíkos** és **memóriakártyás** eszközök és rendszerek egyedülálló hazai választékát kínáljuk.
Új fejlesztésű berendezéseinket az idei **COMPFAIR** kiállításon is megtekintheti.

Addig is bővebb felvilágosítás:

**IDENTIK Elektronikai Kft.**
1143 Budapest XIV., Cserei u. 6.
Telefon: 252-7524, 183-4106 Telefax: 252-7524

INFORMÁCIÓKÉRÉS: 17 ▲

# MAXELL FLOPPYLEMEZEK ŐSZI VÁSÁRA!!

| (Árak: Ft/doboz) | 1–10 doboz | 11–50 doboz | 51 doboz felett |
|---|---|---|---|
| MAXELL 5.25" DS/DD | 700 | 670 | 640 |
| MAXELL 5.25" DS/HD | 990 | 960 | 920 |
| MAXELL 3.5" HD | 1800 | 1700 | 1650 |

## Földelhető polarizált üvegszűrők bevezető áron!

| 1–10 db | 11–50 db | 51 db felett |
|---|---|---|
| 2250 Ft/db | 1690 Ft/db | 1990 Ft/db |

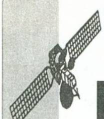
***ÁRAINK A 25% ÁFA-T MÁR TARTALMAZZÁK!***

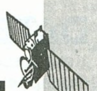
**UNITRADE**
*Szervezési, kereskedelmi és Számítástechnikai*
**K.F.T.**

1073 Budapest VII., Erzsébet krt. 48.
Telefon/Fax: 142-2115

**...nemcsak számítástechnika**

# HETENTE FÖLDKÖZELBEN





*a* **TELEHOLD**

ÖN A LEGTÖBBET KAPJA,
ha megrendeli a hazánkban fogható valamennyi fontos műholdprogram legrészletesebb műsorfüzetét.
Ingyenes hirdetési lehetőség,
a hazai és a környező országok tévéműsorai.

Keresse szerdától az újságárusoknál vagy fizessen elő!

*Egy évre előfizető olvasóink 12 héten át ingyen kapják*

*a* **TELEHOLD***at!*

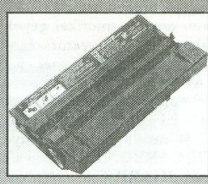
INFORMÁCIÓKÉRÉS: 19 ▲

# TONER KFT

**Másológép és lézerprinter-kazetták felújítása német technológia alapján, garanciával, feketében és színesben.**

## NE DOBJA EL!

**Felújítható kazettatípusok: FC, PC, EP, EP-S, EP-L, SHARP Z-30, Z-50.**



**1095 Budapest Mester u. 21. Tel.: 113-1687 Tel./Fax: 134-3516**

Europa International

# Szakértőrendszer-projektek és -alkalmazások

Ebben a hónapban rovatunkban — az előző cikkek információira is támaszkodva — konkrét munkákat mutatunk be, melyek a „tudás" alapján születtek és élnek is. Bevezetésként áttekintjük — noha már volt erről is szó, de nem ilyen tömör és „ismétlő" gondolatmenetben — az ez irányú fejlődés és a jelen legfontosabb mozzanatait, természetesen elsősorban hazai vonatkozásokban.

A tudásalapú/szakértő rendszerek fejlesztésére és futtatására alkalmas keretrendszerek, shellek piaca az 1980-as évek elején indult be. Kezdetben általános célú (ún. horizontális) shelleket építettek — először csak nagygépeken, majd (1986-tól) PC-ken is.

Ezek főleg szabályalapú rendszerek voltak, de több frame-alapú, ill. több tudásreprezentációt támogató (ún. hibrid) eszköz is volt köztük. Az ezután beindult „shellháború" a minél többféle szolgáltatással, „jó tulajdonsággal" rendelkező eszközök kifejlesztésének irányába hajtotta a fejlesztőket: bizonytalanságkezelés különböző módjai, kevert (előre- és hátrafelé haladó) szabályvégrehajtás, menük-ablakok, „nem tudom" válasz kezelése, többféle tudásreprezentáció támogatása, külső nyelvekkel és adatbázisokkal való kapcsolat biztosítása stb. A shellalkalmazók, vagyis az egyes szakértő rendszerek fejlesztői körében egy technológiavezérelt, prototípus-központú szemléletmód terjedt el; ez egyik oka annak, hogy legtöbb fejlesztés megállt a prototípus szintjén.

1988 végére az alkalmazók kezdeti lelkesedése alábbhagyott; a shellpiac időlegesen megtorpant, „befagyott".

Az előző számban is vázoltuk már, hogy 1989-től a következő két új irányba mozdult el a tudásalapú technológia:

— (az addiginál szervesebb) integrálás a hagyományos (például adatbázis-) technológiákkal és újabb (egyéb MI-, hipertext- stb.) technikákkal, valamint

— (az addigi általános célú eszközök mellett/helyett) problématípushoz szabott, alkalmazásorientált (ún. vertikális) eszközök kifejlesztése és bevezetése.

Napjaink sürgető feladata a (még „éretlen") tudásalapú technológia és (a korántsem hibamentes szoftvereket produkáló) egyéb technológiák (eszközök + módszerek) flexibilis integrálása.

## Az utóbbi tíz évben

A magyar szakemberek kezdettől fogva nyomon követték az MI-kutatásokat. Az első, nagygépes shellekhez ismert okokból nem volt annak idején hozzáférésük, de sok PC-s shellhez hozzá tudtak jutni (legálisan is), sőt, ilyeneket hazai intézményekben is fejlesztettek. Az ismertebbek (és a fejlesztő intézmények):

— általános célú termékek: ALL-EX/3 (All/Multilogic), Genesys (Számalk), MProlog Shell (IQ Soft)

— alkalmazásorientált eszközök: KAS (MTA SZTAKI), Metabolexper (CompuDrug), Pangea (BME), Realex (BME), OPSQL (MTA KFKI), Zexpert (IQ Soft).

1985-91 között sok hazai tudásalapúrendszer-projektet műveltek Magyarországon (részben OMFB-MTA támogatással). Az első hazai szakértőrendszer-projektek orvosi, majd kémiai, építészeti és számítástechnikai témákban voltak. Jelenleg előtérbe kerültek az ipari, pénzügyi és közgazdasági témák; egyre több hoz létre (adatbázissal) integrált alkalmazást.

A kezdeti időszakban, a technológiával ismerkedvén, rengeteg prototípus készült. Sajnos sok a kísérletirendszer-fázist elért, de megfelelő pénzügyi háttér és emberi érdekeltség híján félbehagyott fejlesztés. A ma is munka alatt lévő projektek közül az ismertebbek — témakörök szerint (projektek száma zárójelben): orvostudomány-egészségügy (12), energetika (6), építőipar (3), egyéb ipari projekt (6), kémia (9), közgazdaság-pénzügy (9), végül, de nem utolsó sorban számítástechnika (4). Összesen 40 intézményben folytak eddig ilyen fejlesztések — további 40 (hazai és külföldi) intézményből bevonva tárgyköri szakértőket a munkába. Jelenleg közel 30 alkalmazás van termékszinten (ennél jóval kevesebb van napi használatban).

## Most és az elkövetkező években

A már említett pénzügyi és emberi problémák mellett további probléma az is, hogy a legtöbb esetben hiányzik az a szervezett információs háttér, amelynek adatai „fölött" lehetne csak igazán hasznosítani egy intelligens tanácsadó rendszert. Így például egy kórházban úgy lehetne igazán hatékonyan működtetni egy diagnózisra/terápiára javaslatot tevő rendszert, ha az a kórházi információs rendszerben tárolt beteg- és kóradatok változását rendre nyomon követi, és ("watch dog" módjára) csak akkor jelez az adott beteg kezelőorvosának , ha valami „újat tud mondani".

A tudásalapú rendszerfejlesztést az eddigiek mellett a tudásalapú technológia „éretlenségéből" fakadó számítástechnikai problémák is nehezítik; néhány ezek közül:



INFORMÁCIÓKÉRÉS: 20 ▼

— Integrált rendszerépítési módszertan hiánya, beleértve a feladattípust meghatározó, ill. feladattípushoz „illő" hatékony megvalósítási technikákat kiválasztó módszerekét.

— A tudásbázisba beépített, a felfrissítések során változó szakmai ismeretanyag verifikálása. A gyakorlat az, hogy átlagos és szélső helyzeteket leíró „esetet" kidolgozásával, azok esetenkénti lefuttatásával, az üzembe állítás előtt pedig különböző felhasználói körök „béta-tesztelésével" támogatják az ellenőrzés-vizsgálatokat (a szisztematikus ellenőrzésre hatékony és általánosan alkalmazható módszert nem is lehet adni).

— A szaktudás „begyűjtésének" problémaköre — ezzel az Alaplap MI-cikksorozata (ez a rovat) részletesebben is fog foglalkozni.

Mindezen problémákat súlyosbítja az, hogy a legtöbb tárgyterületen nem alakult még ki az alapfogalmak és kapcsolatok egységes rendszere (taxonómia, „nevezéktan"). Rendszerint minden projekt saját maga „gyűjti be" és rendszerezi a számára aktuális ismeretanyagot, ami azon felül, hogy megtakarítható lenne, a rendszerek adaptálásánál külön nehézséget okoz (nem véletlen például, hogy az USA-ban jelenleg dolgoznak egy egységes orvosi nyelvi rendszeren).

Nincs kialakult mechanizmus a szakértőrendszer-alkalmazások termékként való értékesítésére sem, márpedig kézenfekvőnek látszik a tudásbázisba (sok munkával) beépített ismeretanyag további hasznosítása. A szakértőrendszer-piacokon a rendszerépítő eszközök vannak jelen (szoftver, dokumentáció és a felhasználót támogató különböző szolgáltatások formájában). Alkalmazói rendszerek helyett is inkább alkalmazás specifikus eszközkészleteket visznek a piacra. Ez azzal magyarázható, hogy az alkalmazások tudásbázisában tárolt (és a feladatmegoldásra a szituációt függően „mozgósított") szakértői ismeretanyagot igazán csak a rendszerépítésnél felkért szakértők respektálják — főleg, ha nagyon sok a nem általánosan szabályozott, heurisztikus ismeretanyag a tudásbázisban (fokozottan bizalmi kérdés a rendszer tanácsait, javaslatait figyelembe venni, hisz mindenki tudja, hogy a döntés felelősségét nem lehet átruházni a gépre). Például egy pénzügyi/befektetési tanácsadó rendszernek jobb a terjedési helyzete, mint egy orvosinak, hiszen az előbbibe foglalt szakanyag egyértelműen — központi szabályozás, közlönyök által — ellenőrizhető. A rendszerek adaptálása tehát jellemzően az e szakértők ismerőseinek környékén történik.

Az egyes tárgyterületeken szükség lenne egy olyan, nagy tekintélynek örvendő és független szakmai fórumra, amely szakmailag minősítené, és „ilyen és ilyen körben alkalmazásra javasolt" záradékkal, valamint hitelesítő pecsétjével látná el az adott szakértő rendszert — orvosi rendszer esetén például az Orvosi Kamara tölthetne be ilyen szerepet (gondoljunk arra, hogy egy „igazi" szakértőt is csak megfelelő ajánlás mellett szokás alkalmazni). Jobb a helyzet akkor, ha a rendszer építését egy ilyen hatókörű szervezet fejleszteti...

Hazánkban egyre több területen foglalkoznak intenzíven — és az egységesség igényével — információs alaprendszerek kifejlesztésével. Várható, hogy a tudásalapú rendszerek iránti érdeklődés is hamarosan felélénkül. (Rovatunk további cikkei és részben A hónap témája is ezt támasztja alá.)

**Sántáné Tóth Edit**

# MProlog-alapon

A Zexpert keretrendszert a legnagyobb osztrák bank, a Zentralsparkasse és a Landerbank házasságából tavaly létrejött Bank Austria számára készítették, illetve jelenleg is fejlesztik...

A Bank Austria több száz bankfiókjában — az egységes, magas szintű tájékoztatást biztosítandó — tanácsadó rendszereket állítanak fel; az egyik ilyen tanácsadó rendszerük segíti például az építési kölcsönök iránt érdeklődő ügyfeleket (és amelyet az IBM ESE nevű, nagygépes shelljében korábban realizáltak). A Zexpert keretrendszert az ezután bevezetendő hasonló rendszerek technológiai alapjául szánják a banknál.

A következőkben szó lesz a fejlesztés előzményeiről, az alkalmazott tudásreprezentáció lényegesebb alapelveiről, a felhasználói interfészről, valamint a keretrendszer fejlesztésének jelenlegi helyzetéről.

## Feltételek és környezet

Mielőtt eldöntötték volna, hogy általános célú termék helyett egy saját célra létrehozott „feladatorientált" szoftverre van inkább szükségük, a bank számítástechnikai szakemberei természetesen több eszközt is megvizsgáltak, elsősorban a következő két szempont figyelembevételével:

— A kiválasztott eszközzel kifejlesztett tanácsadó rendszereknek teljes mértékben illeszkedniük kell a bank szabványaihoz. Nyilvánvaló, hogy egy nagy bank számítástechnikai életében a központi szerepet a „klasszikus" banki szoftverek játsszák, amelyek képernyőképeinek, dokumentációinak stb. már hosszú évek alatt kialakultak a (belső) szabványai. A bevezetendő tanácsadó rendszereknek ezekhez kell alkalmazkodniuk.

— A szóban forgó tanácsadó rendszereknek viszonylag gyorsan változó tudás alapján kell működniük, hiszen egy pénzügyi szabályozót, döntést sokkal könnyebb megváltoztatni, mint például egy természeti törvényt. A bank úgy akarja teljes mértékben kézben tartani tanácsadó rendszereit, hogy a rendszerben tárolt tudást a bank szakértői át tudják tekinteni — el tudják olvasni, és könnyen, „természetes nyelvet" használva módosíthassák. Ez a legfontosabb követelmény a tudásreprezentáicóra nézve.

A vizsgálódások ahhoz a döntéshez vezettek, hogy ezeket a szempontokat a bank egy új, saját eszköz kifejlesztésével tudja legjobban érvényesíteni. Azt is elhatározták, hogy az új rendszert MPrologban fogják megvalósítani; ehhez az alapot a bank munkatársai által elvégzett modellkísérletek adták. (Ennek egyik konkrét eredménye volt 1990-ben egy Zexpertnek nevezett szakértőrendszer-keret prototípus.) A bank és az IQ-Soft (akkor még: az SZKI Elméleti Laboratóriuma) kapcsolata itt kezdődött: a bank a fenti kísérletekhez

megvásárolta az MPrologot, és így programozási munkájuk során kapcsolatba kerültek a magyar MProlog-fejlesztő csoport tagjaival.

Az első Zexpert prototípus még IBM nagygépen futott, azonban hamar eldőlt, hogy az új rendszert PC-s alapon kell megvalósítani. A fejlesztés elején még DOS-ban kellett dolgozni, majd — a Bank Austria új számítástechnikai stratégiájának megfelelően — az OS/2 operációs rendszerben (az MProlog alaprendszer biztosítja a termék „hordozását").

## Kevert tudásreprezentáció

A Zexpert ún. kevert tudásreprezentációt alkalmaz; a tárgykör objektumait paraméterekkel és frame-ekkel ábrázolja, míg a paraméterek és a frame-ek slotjai (?!) közötti összefüggéseket szabályok rögzítik. Utóbbiakat hátrafelé haladó következtetési mechanizmus működteti.

A tudásreprezentáció részletes ismertetése helyett itt most csak néhány olyan alapelvet vázolunk, amelyek a tudásbázis „olvashatósága" és áttekinthetősége irányába mutatnak (és amelyek lehetővé teszik, hogy a banki szabályozókat ismerő bankember közvetlenül tudja frissíteni, módosítani a tudásbázist):

— Korrekt német nyelvű szabályok.

A Zexpertben külön hangsúllyal szerepel a korrekt német nyelvű szabályleírás lehetősége (amely más, például a magyar nyelvű szabályok megfogalmazásának lehetőségét is biztosítja). Ez elsősorban a szabályok feltétel-, illetve következményrészében a német nyelv szabályainak megfelelően, különböző szórenddel megfogalmazott összefüggések egymáshoz kapcsolását jelenti.

— A deklaratív és az algoritmikus tudás szétválasztása.

Szabályalapú rendszereknél gyakori, hogy „átesnek a ló másik oldalára", és a jellegénél fogva algoritmikus tudást is (mint például a tanácsadó rendszer fő vezérlési folyamata: hol zajlik a párbeszéd, hol történik az eredmények kinyomtatása stb.) deklaratív módon fogalmazzák meg, rosszabb esetben el sem különítve a tárgyköri ismeretanyagtól. Ez a banki szakértő számára nehézzé tenné a rendszer áttekintését. A probléma kikerülésére a Zexpertnél kiemeltük a tudásbázisból, és külön rendszerkomponensként realizáltunk egy ún. vezérlő részt.

— Bemenő adatok ellenőrzése.

Az bemenő adatok ellenőrzéséhez speciálisan kezelt szabályok adhatók, így elérhető, hogy csak helyes adatok kerüljenek a tudásbázis dinamikus részébe.

— Szövegtípusú paraméterek.

A felhasználónak megmutatandó szövegek is szabályokkal állíthatók elő, úgynevezett szövegdirektívákkal előírva a formátumozást.

További fontos elv volt még egyebek között a tudásbázis világos szerkezetét, így biztonságos karbantartását biztosító modularitás, valamint a szerepköröket kifejező típusok bevezetése (az MProlog, mint a legtöbb Prolog nyelv, típustalan nyelv: nem lehet benne adattípusokat deklarálni).

## Felhasználói interfész, fejlesztő és futtató rendszer

A felhasználói interfész már a DOS-os változatban is a CUA- (Common User Access) szabványt követte, noha itt a hangsúly még az interfész specifikációján, illetve annak kipróbálásán volt.

A CUA az IBM által kifejlesztett új rendszernek, az SAA-nak (Systems Applications Architecture) egyik fő eleme: korszerű alkalmazások kifejlesztését és végrehajtását biztosító felhasználói interfész, konvenciók és protokollok együttese.

A fejlesztő környezet központi része egy tudásbázis-dedikált editor: a Zexpert szelekciós mechanizmusával — a fejlesztés fókuszaként — a tudásbázis tetszőleges része kiválasztható. Az egyes fejlesztési funkciók az így kiválasztott részre fognak alapértelmezésben végrehajtódni.

A fejlesztés eredeti célkitűzéseihez kapcsolódik a karbantartás támogatása: a rendszer felhasználóbarát szolgáltatásokat nyújt azoknak a banki alkalmazottaknak, akik valamilyen változtatást akarnak elvégezni a (mások által kifejlesztett) tanácsadó rendszeren.

A rendszer áttekintését különböző kereszt-referenciák, illetve a függőségi fák segítik. Ezek az interaktívan kezelt fák azt mutatják meg, hogyan függenek össze a rendszer különböző objektumai.

A rendszer logikájának megértését segíti a nyomkövető, illetve a magyarázatadó komponens. A Zexperttel dolgozó fejlesztő többszintű, pozitív és negatív „hogyan" és „miért" magyarázatot is kaphat.

Az előbbi interaktív szolgáltatások mellett egy batch üzemű, „visszajátszó" teszt is fontos szerepet jászik a rendszerváltoztatások letesztelésében. A párbeszédek során tárolhatók a felhasználói válaszok és bizonyos kielölt eredmények, illetve részeredmények. Később bármikor vissza lehet játszani e párbeszédeket, melynek során a Zexpert automatikusan elvégzi az adott párbeszédhez tartozó következtetést, összevetve a kapott (rész)eredményeket azok elvárt/tárolt változatával.

A fejlesztő rendszertől elkülönül a futtató rendszer. Természetesen a fejlesztő rendszerből hatékony és kompakt futtatható kódot lehet generálni; így a végfelhasználó már nem találkozik a Zexpert rendszer teljes fejlesztői funkciókészletével, csak a számára szükséges (és megengedett) futtató és fejlesztő funkciókkal.

A fejlesztő rendszer első, DOS-ban megvalósított változata 1992 elejére készült el; e munkával párhuzamosan folyik az OS/2-s változat fejlesztése, amely ez év őszén fejeződik be. A banknál jelenleg már „élesednek" az Zexpert első valós alkalmazásai.

**Farkas Zsuzsa**

# Az EXPO kockázata

**A világkiállítással kapcsolatban talán a legtöbbször elhangzó kifejezés a „kockázat" — a legkülönbözőbb szövegösszefüggésekben. Mindenesetre a kockázat nagyságát latolgatják az előkészítésében részt vevő szakemberek is.**

A Világkiállítási Programiroda által felkért szakértők a világkiállítás kockázati elemeit az alábbi csoportokba sorolták: társadalmi, politikai; gazdasági, pénzügyi; műszaki, megvalósíthatósági-üzemeltetési; jogi, szabályozási; szervezeti, szervezési és egyéb, a fentiekbe nem besorolható kockázatok. Az így összeállított lista 179 kockázati elemet tartalmaz.

A világkiállítás esetében véleményünk szerint nem voltak alkalmazhatók a megszokott döntéselméleti, kockázatelemzési módszerek. Egyes dimenziókban természetesen jogos e módszereknek a felhasználása, de a világkiállítás megrendezésére vonatkozó egyedi döntés összes kockázati hatásának összegzését egyetlen általunk ismert integratív jellegű modell sem lett volna képes megbízhatóan elvégezni. Ezért választottuk a szakértő rendszert mint megoldást.

Ami a gazdasági hátteret illeti, egyszerűen összefoglalva: a gazdaság normál (piacgazdasági irányba mozduló) viselkedése a feltétele annak, hogy a világkiállítás kapcsán

— privatizációs többletbevételek realizálódjanak;

— sok új munkahelyet lehessen teremteni;

— fellendüljön az idegenforgalom;

— meginduljon egyes nem építőipari és idegenforgalmi ágazatok párhuzamos fejlődése (multiplikátor hatás);

— elkerülhető legyen az import növekedése;

— lehetővé váljon a megfelelő mértékű külföldi tőkebevonás, és ezáltal egy közép- és hosszú távú pozitív hatás lehessen az eredmény.

A kockázat csökkenésének irányába mutató megoldás a világkiállításhoz kapcsolódó rendezvények szervezése, a vidék bekapcsolása. Ez azonban csak a kormány, az államigazgatás és az önkormányzatok megfelelő együttműködése és érdekazonossága mellett valósítható meg. A világkiállítás új koncepciójában már egyértelműen elkülönül az infrastruktúra fejlesztése.

## Hihetetlenül érzékeny költségvetés

Kiemelten kellett foglalkozni a világkiállítás pénzügyi, finanszírozási kockázatával. Az az igen érdekes helyzet alakult ki, hogy még ha a világkiállítás egyenlege pozitív is lenne, illetve a kiadási oldalt sikerülne megbízható ajánlatokkal lefedezni, ez a költségvetés kockázatát nem befolyásolja. Ez azért van így, mert a költségvetés kockázata elsősorban a vállalt garancia nagyságától függ. A költségvetés szempontjából legrosszabb esetet sokkal nagyobb súllyal vettük figyelembe, mivel a magyar költségvetés jelenleg a kockázatra hihetetlenül érzékeny.

Az elkészült, közel sem teljes tudásbázisú szakértői rendszer végül is mintegy 120 szabályt tartalmazott. A megvalósítást az ALL/Multilogic által kifejlesztett ALLEX PLUS keretrendszerben végeztük OS/2 operációs rendszer alatt.

Az ALLEX PLUS egy frame-/szabályalapú szakértőrendszer-keret, amely elsősorban visszafelé következtető, „mélységben először" mechanizmussal rendelkezik, de az öröklődés lehet „széltében először" kapcsolatú is. Bizonyos mértékig lehetőség van az akciók felhasználásával előrefelé következtetésre is, de ilyenkor ehhez programozási trükkök szükségesek. Két tipikus szabály — példaként előbb természetes nyelven, majd ALLEX PLUS-beli alakban:

„Az EXPO korlátozottan valósítható meg, ha az image teremtése közepes, katalizáló hatása nincs, környezeti hatása semleges, finanszírozhatósága ugyan megoldott, de attraktivitása közepes."

„A népgazdaság külgazdasági egyensúlya romlik, ha az import meghaladja a tőkebevonás mértékét, nem érvényesül a kínálatbővítő gazdaságpolitika, és a forráskivonás meghaladja az addicionális forrásokat."

```
if expo^image_teremtése is
   közepes and
   expo^katalizáló_hatás
  is nincs and
   expo^környezeti_hatás
  is semleges and
   expo^finanszírozhatóság
  is megoldott and
   expo^attraktivitás is
  közepes
        then
   expo^megvalósíthatóság
  is korlátozott
if not(népgazdaság^
tőkebevonás is meghaladja)
and
   not(gazdaságpolitika^ki
  nálatbővítés is
 érvényesül)
and
   not(népgazdaság^addicio
  nális_források is
 megha
 ladja)
        then
if népgazdaság^
```

`külgazdasági_egyensúly is romlik`

A fentiekből látszik, hogy az „if then" alakú szabályok a „frame^attribútum is érték" típusú fogalmakon vannak értelmezve.

## A bizonytalan tudás kezelése

Külön figyelmet érdemelhet a kockázat számszerűsítésének megvalósítása. Nyilvánvaló, hogy az lenne a jó, ha a különböző EXPO-megvalósításokhoz valamilyen „bizonytalansági" faktort tudnánk hozzárendelni. Közismert, hogy egyetlen általánosan elfogadott algoritmus sincs a bizonytalan tudás kezelésére. Az általunk választott módszer a következő volt.

A szakértők által legfontosabbnak tartott 70 kockázati tényezőhöz pontértékeket rendeltünk. A konzultáció során az egyes kérdésekre adott válaszoktól függően a kockázati tényezők fennállnak vagy nem, és így rontják vagy javítják a világkiállítás megvalósíthatóságának pontértékét. Minél magasabb pontszámot kap egy alternatíva, annál kisebb a megvalósításának kockázata. A konzultáció kezdetén az induló pontérték 1000. Az egyes kockázati elemek „osztályokba" vannak sorolva, mely osztályok pontértékei: image – 100, katalizáló hatás – 200, finanszírozhatóság – 300, attraktivitás – 200, környezeti hatás – 200. Ezen belül például a katalizáló hatás 10 tényezőtől (attribútumértéktől) függ (igen/nem): munkahelyeket teremt +/-20, a tőkebevonás mértéke meghaladja az importot +/-30, ...jó a piacfeltáró munka +/-10, jó a marketingmunka +/-10.

Az összesítésnél a Világkiállítás megvalósítása lehet kockázatmentes (1401—1800), kevéssé kockázatos (1201—1400), vállalható kockázatú (1001—1300), nagyon kockázatos (901—1000), lehetetlen (0—900).

A pontszámítást a démon mechanizmuson keresztül valósítottuk meg. Mint ismeretes, a démonok olyan aszinkron tevékenységeket megvalósító elemek, melyek figyelik, hogy egy attribútum mikor kap értéket, és ennek megfelelően végrehajtanak egy előre meghatározott tevékenységsorozatot. A mi esetünkben a tevékenységsorozat a pillanatnyi pontérték megfelelő mértékű változtatása volt.

Például a tőkebevonás mértékét (népgazdaság^tőkebevonás) attribútumot két démon figyelte, az egyik akkor indult be, ha a tőkebevonás mértéke meghaladta az importot (+30 pont), a másik pedig akkor, ha a tőkebevonás mértéke nem haladta meg az importot (-30 pont).

**Futó Iván**

# Turbógenerátor-rezgésdiagnosztika

A beruházási források szűkössége, a működési költségek csökkentésének igénye, a gazdasági és emberi szempontból egyaránt biztonságos üzemvitel követelménye, továbbá az elmúlt évek néhány katasztrófája ráirányította a kutatók figyelmét a technológiai kulcselemek üzemben tartásával és karbantartásának tervezésével kapcsolatos kérdésekre.

A Paksi Atomerőmű részére az üzemi turbógenerátor-rezgésdiagnosztikai szakértői rendszer (TDSZR) 1990 végén készült el egy OKKFT-finanszírozású, 3 éves program eredményeként. 1991-ben egy hasonló rendszert helyeztünk üzembe a Kalinyini Atomerőműben (Oroszország) is.

A TDSZR a Számalkban kifejlesztett Genesys 2.1 shell erősen módosított változatával készült. A módosítás elsősorban a futtató modult érintette, valamint az egész rendszer egy önálló eredményközlő modult kapott. Az eredményközlő modul úgy készült, hogy a vizsgált turbina állapotát leíró diagnózist (egy kinyomtatott jelentés formájában) a nem szakértő felhasználó (átlagos erőművi operátor és/vagy karbantartó) is elő tudja állítani és megérteni. A szakértő pedig a diagnózisok hátterét is megismerheti.

A Genesys 2.1 TurboPROLOG, az eredményközlő modul és a kiegészítő, adatolvasó, mérő eljárások C nyelven íródtak. A TDSZR rezgésdiagnosztikai szakértő. Ez azt jelenti, hogy az ismeretbázisban rezgésjelenségekre vonatkozó szabályok vannak. A kiértékelés során a rendszer nem kikérdezi a felhasználót, hanem az ARGUS rezgésmonitorozó rendszerrel periodikusan mért adatokból és a TDSZR-hez fejlesztett karbantartási adat- és eseménytár adataiból dolgozik. A mért adatok között a csapágyakon mért rezgésjelek (integrális értékek és spektrális összetevők) és a rezgésképet befolyásoló legfontosabb technológiai változók (teljesítmény, hőmérsékletek stb.) találhatók.

## Az ismeretek

Az ismeretbázis feltöltésében a VEIKI és a PAV szakemberei vettek részt. A rendszer verifikációja és validációja elsősorban tapasztalati úton és kísérletekkel történt, kisebb részben egy rotormodell alkalmazásával. A szabályok a turbógenerátor technológiailag elkülöníthető fő részeire, azaz két csapágyon megtámasztott forgórészekre, illetve összekapcsolt rotorpárokra vonatkoznak. A rendszer jelenleg 13 különböző, a fő részekhez rendelt hiba azonosítására képes, úgymint a kiegyensúlyozatlanság különböző típusai, csapágy- és csapágytámasz-rezonanciák, lazulások, tengelygörbeség, rotorrepedés, egytengelyűségi hiba, a jelenlegi ismeretekkel még nem kezelhető rezgésképváltozások megjelenése.

Az ismeretbázis kb. 600 feltételt (200 szabályt) tartalmaz. A rendszer turbinánként 24 rezgésjelet (a spektrális komponensekkel és kiegészítőkkel ez összesen 480 bemenő adat), mintegy 10 fontos technológiai változót, és néhány tucat karbantartási eseményt kezel.

## A biztosítékok

Ipari környezetben különösen fontos a diagnózisok, illetve hibajelzések megbízhatósága. Ennek érdekében a mérési adatokat szolgáltató ARGUS rendszer hihetőségvizsgálatán túl további jelellenőrzéseket is végzünk a kiértékelés előtt. A TDSZR diagnózisaihoz mindemellett bizonyossági tényező tartozik. A 0 és 100% közötti szám a szóban forgó hiba fennállásának bizonyosságát mutatja. A shellben többféle szabálykiértékelési algoritmus van, erősítő, súlyozó, általános. Ezek kombinációjával lehetett elérni, hogy az eredő bizonyosság függjön attól, hogy a szakértő a tapasztalatai alapján mennyire tart „erősnek" egy szabályt, és attól, hogy az ugyanarra a hibára mutató vagy az ezt kizáró különböző feltételek közül hány teljesül (erősítés, gyengítés).

A TDSZR az elmúlt egy évben jelentősen tehermentesítette a szakértőket és diagnosztikusokat. Téves diagnózist nem adott, jelentései összhangban vannak a turbinák tényleges állapotával. Jelenleg az ismeretbázis további bővítése folyik.

**Bessenyei Zoltán—Tomcsányi Tamás**

## És Rothschild nyert...

# Az információ mozgatása

Sokat hallunk a számítógépekről, gyakran látjuk őket, mert napjainkra mindenki számára érdekessé váltak. A témával foglalkozó előadások, cikkek, könyvek jó része a szenzációsabb tálalás érdekében gyakorta kihasználja a laikusok tájékozatlanságát. Ez írásommal a legfrissebb újdonságok ismertetése helyett a számítástechnika lényegét és a civilizációban elfoglalt helyét próbálom körvonalazni — remélve, nem csakis „újszülötteknek" hasznosan.

Amióta csak eleink néhány millió évvel ezelőtt lejöttek a fáról (ha fent voltak egyáltalán), alapvetően két problémát kellett megoldaniuk mindennap: különböző anyagok összegyűjtését, szállítását, feldolgozását és tárolását, valamint különböző információk összegyűjtését, szállítását, feldolgozását és tárolását.

Úgy tűnik, a kettő közül az anyagokkal kapcsolatos feladatok megoldásában elért eredmények voltak eddig meghatározóak. Erre utal az ősidők kőkor, rézkor, vaskor szerinti felosztása, a kerék feltalálásának jelentősége (az amerikai kultúrák nem ismerték a kereket!), a hajózás, a vasút, végül századunkban az autó és a repülés mindennapossá válása.

Az információ valahogy nem látszott ilyen lényegesnek. Átfogó értelemben ez új fogalom; fontosságában azonban semmivel sem maradt el az anyagok mögött, már az emberiség hajnalán sem.

### Varázslók, hűbérurak és pénzemberek

Előfordult például, hogy a horda sámánját — aki tudta, hogyan kell tüzet gyújtani — megette a kardfogú tigris. Amennyiben a módszert balesete előtt nem tanította meg valakinek (egyébként bolond lett volna, hiszen ezzel saját fontosságát és tekintélyét felezi meg), ez az információ elveszett a csoport számára. A veszteség viszont nemcsak az étlapot tette átmenetileg egyhangúbbá, de télen súlyosabb következményekkel is járhatott...

Ebben a korban a lassú fejlődés magyarázata az információtárolás és -terjedés szinte teljes hiánya. Mindent csak szájról szájra adhattak tovább, és ha a lánc megszakadt — márpedig ez a folytonos létbizonytalanság miatt gyakran megtörtént —, újra fel kellett a dolgot találni, vagy megszerezni más csoportoktól.

Az írás volt az első — máig nagyon fontos — információtárolási mód. (A legtöbb uralkodó büszke volt könyvtárára, még ha maga nem is tudott olvasni.) Az írás megjelenésével az információ gyors és biztonságos továbbítása vált a legnehezebb feladattá. Sok mindent használtak az idők során: dobokat, futárokat, postagalambokat. Az állam megszilárdulásával megszervezték a postaszolgálatot.

### Rothschildnak megérte

A megfelelő információ — akár konkrét pénzbeli — értéke közismert lett.

Rothschildnak megérte, hogy saját futárt bízzon meg, értesítse őt — egyenesen a londoni tőzsdén! — a waterlooi csata kimeneteléről. (A vállalatok akkor még nem voltak multinacionálisak, így angol győzelem esetén az angol vállalatok előtt nyílik meg az európai piac, ellenkező esetben a francia cégek aratnak.) Rothschild embere kb. negyedórával előzte meg a hivatalos királyi futárt. A taktikus pénzember a hír hallatán árulni kezdte angol részvényeit. (A tőzsdén előzőleg feszült volt a hangulat, alig kötöttek üzletet. Egyszercsak egy ember potom áron kínálja az angol részvényeket...) Kitört a pánik. Mindenki szabadulni igyekezett az angol papíroktól, amelyek ára percek alatt lezuhant. Rothschild ügynökei az ő egyetlen rejtett intésére annyit vásároltak fel, amennyit csak bírtak. Hamarosan megérkezett a királyi futár is: az angolok győztek!

### Analógia vagy „párhuzam"?

Az elektromosságot a távíró és a telefon által kezdték kihasználni az információ továbbítására — így az elvileg maximális, vagyis a fénysebességet érték el. A legfontosabb különbség a számítógép és mondjuk a rádió információtovábbítása között az, hogy a rádió analóg, a számítógép pedig digitális jelekkel dolgozik.

Ennek megvilágítására vigyünk végig egy gondolatot. Nos, azt a feladatot kapjuk, hogy létesítsünk összeköttetést két város között — nagy mennyiségű információ gyors, biztonságos és lehetőleg olcsó küldésére. Kifeszítünk hát két szál drótot a két helység között, amelyeken elektromos feszültséget tudunk továbbítani. Az információt számok formájában kapjuk. Ha el kell küldenünk például a 20-at, 20 voltot kapcsolunk a vezetékre, amit a túloldalon megmérnek és lejegyeznek.

Gondjaink akkor kezdődnek, ha például a 20 000 számot kell továbbítanunk, mert ekkor már többek között az életvédelmi szempontokat is figyelembe kéne venni. A módszer többféleképpen is toldozható-foldozható, de igazán olcsó megoldást akkor kapunk, ha csak kétféle feszültséget engedünk meg a vezetéken: mondjuk 0 és 5 voltot. Ekkor csak kétféle számjegyet tudunk továbbítani: a 0-át és az 1-et. A küldendő számokat ezért átváltjuk kettes számrendszerbe, mert ebben csak kétféle számjegy (a 0 és az 1) fordul elő, így

bármilyen nagyságú számot egyszerűen el tudunk küldeni. Ezzel "feltaláltuk" a digitális információtovábbítást. Az így kódolt adatokat nemcsak továbbítani, de feldolgozni is — például két számot összeadni — jóval könnyebb.

A módszert egyszerűsége és olcsósága miatt számos területen használják, a karóráktól az orvosi műszerekig.

## Repülőgép és kamion helyett/mellett

A számítógépek mindennapjaink részévé váltak. Első pillantásra ez csupán a technikai fejlődés egy fokozatának látszik, hasonlóan ahhoz, ahogyan a korábban csakis a stúdiókban használt hifi- és videoberendezések bevonultak a családi otthonokba.

Valójában lényegesen többről van szó. Századunkban az emberek százmillióit sikerült még ötven évvel ezelőtt is elképzelhetetlen életszínvonalra emelni, ami persze nem vigasz azoknak, akiknek az élete alig változott az utolsó ötszáz évben. Ezzel együtt a fejlődés — sőt, a szinten tartás — korlátai is egyre világosabbá váltak: a kifogyó energia- és nyersanyagkészletek, valamint a globális méretű környezetszennyezés. A modern ipari társadalmakban az ember szinte minden tevékenységével hozzájárul ezeknek a gondoknak a súlyosbodásához, az autóhasználattól kezdve az eldobott műanyag palackokig. A károk csökkentésére természetesen voltak és vannak erőfeszítések, a kisebb fogyasztású autók kifejlesztésétől a hulladékok újrafeldolgozásáig.

Ezek azonban — bár nagyon fontosak — cseppek a tengerben, a katasztrófa elkerülésére csak egy új elv következetes megvalósításával van esély: az információt kell mozgatni (az emberek vagy az anyagok helyett!) minden olyan esetben, amikor csak lehetséges.

Mit jelent ez a gyakorlatban? Nagyon egyszerű példákkal megvilágítható a lényeg.

## Irány: az informatikai társadalom

A könyvelőnek felesleges mindennap bemennie a céghez, ha a munkájához szükséges adatokat az otthoni számítógépén is megkaphatja, és utasításait otthonról is beküldheti a központba. Ennek eredményeként eggyel kevesebb autó vánszorog át nap mint nap a városon.

Számítógéppel figyelve egy áruház forgalmát, és nyilvántartva a készletet minimálisra csökkenthető az ellátáshoz szükséges fuvarok száma.

Az Egyesült Államokban már adnak ki telefonkönyvet, jogszabálygyűjteményt, cégnyilvántartást, lexikont is számítógépes adathordozón. Előbb-utóbb megjelennek olyan könyvesboltok és könyvtárak is, ahol a legújabb krimit vagy a több száz oldalas regényt egy mellényzsebben elférő mágneslemezre másolják, természetesen nem ingyen. Ezzel erdők maradnak meg, a szállításról és a papírgyárak szennyezéséről már nem is beszélve. Nyilvánvalóan a mágneslemez előállítása is anyagot és energiát fogyaszt, de lényegesen kevesebbet; és mivel egyszerűen letörölhető, nagyon sokszor újra felhasználható.

## Új varázsszó a hálózat

A fenti elvet követő társadalmat szokás informatikai társadalomnak nevezni. Kialakulását természetesen sok ellentmodás kíséri, az említett könyvesboltok megjelenését például valószínűleg nem fogadják majd osztatlan lelkesedéssel a nyomdászok.

A számítógép használhatóságát megsokszorozza, ha összekapcsoljuk más számítógépekkel, így ugyanis nemcsak a saját, de a többi gépen tárolt információhoz is hozzáférhetünk igen rövid idő alatt. A számítógépek összekötésével új varázsszó jelenik meg, a hálózat. A példánkban említett könyvelő otthoni számítógépe hálózatba van kapcsolva cége központi gépével.

A számítógép-hálózat a jelenleg alkalmazott leghatékonyabb információtovábbítási módszer, az informatikai társadalom alapja, miként a vasút és később az úthálózat az ipari társadalom alappillére volt. Számos területen alkalmazzák már ma is, a repülőgépek helyfoglalási rendszerétől kezdve a bankok közötti átutalásokig.

## Világ(hálózat)szerte

Kialakulásának kezdete az ötvenes évekre nyúlik vissza, amikor még lényegesen kevesebb gép működött világszerte. Az Egyesült Államokban a "Szputnyik-sokk" után a „miért?"-et vizsgáló bizottság jelentésének címlapján egy szám volt: 0,4.

A kommunikáció hiánya miatt ugyanis ekkoriban átlagosan minden újítást 2,5-szer fejlesztettek ki az USA-ban. Ennek orvoslására találtak ki egy központi, számítógépen létrehozott nyilvántartást — mai nevén adatbázist —, amelyben folyamatosan vezették, hogy kik milyen témában kutatnak. A szakértők természetesen számítógépet használtak az adatok lekéréséhez, létrejött az első hálózat. (A Szovjetunióban jóval olcsóbban oldották meg a dolgot: az azonos témával foglalkozók egy telepen laktak...)

Végül egy olyan kérdésre is érdemes kitérni, amely időről időre felbukkan a rendszerint laikusoknak szóló irodalomban. A jóslatok szerint a számítógép képességei előbb-utóbb meghaladják az emberi agy képességeit, feleslegessé teszik az embert. Nos, a ma létező számítógépek erre biztosan alkalmatlanok. Egyszerű műveleteket tudnak csak emberi léptékkel mérve nagyon gyorsan végrehajtani. Ezekből a műveletekből sok magasabb rendű feladat is felépíthető, például egy egyenletrendszer megoldása, adatok osztályozása, keresése stb. Azonban sok olyan feladat, amely az ember számára természetes, megoldhatatlan problémát jelent a gépnek. Ilyen például egy térbeli alakzat felismerése, amelynek gépi megoldásáról egyelőre le kellett tenni. Az agy — már amennyit tudunk róla — valószínűleg máshogyan szerveződik. Erre utal, hogy hatalmas tartalékokkal — redundanciával — rendelkezik, viszonylag súlyos sérülés után is funkcionál, míg egy számítógépben egyetlen elem (tranzisztor) meghibásodása is nagy valószínűséggel helytelen működést okoz.

Napjainkban is folyik a vita arról, hogy legalább elvileg szimulálható-e, utánozható-e az ember gondolkodása a jelenlegi számítógépes eszközökkel. Tulajdonképpen az emberi gondolkodás definíciójában a tudorok nem tudnak megegyezni, vagyis hogy milyen feltételeknek kell eleget tennie bárminek ahhoz, hogy kijelenthessük: képes az emberiével egyenértékű gondolkodásra. (Való igaz, hogy egyáltalán a gondolkodásképességet sem mindig egyszerű megítélni — néha embertársainknál sem...)

## Áldás vagy átok? — Számítógép!

A számítógép nem manna, önmagában még megváltást sem hoz, de ellenségnek semmi esetre sem tekintendő — noha vissza is lehet élni vele. (Mivel nem?) Egy eszköz; sok szempontból az egyetlen lehetőség bázisa ahhoz, hogy fajunknak hosszabb időn át is elviselhető körülményeket teremtsünk az egyre kisebbé váló Földön.

**Csórián Sándor**

# IBM PC

# SOLARSOFT KATALÓGUS



**Lemezszám:** 512

**Név:** **STUPENDOS 1.61 & PKZMENU 1.04**

**Szerző:** Douglas Hay & Phil Katz

**Leírás:** PKWare DOS shell és menüvezérelt PKUNZIP.
Phil Katz és csapata nem állt meg a PKZIP és PKUNZIP, valamint a PKLITE programok kifejlesztésénél. Megjelentették STUPENDOS néven DOS shelljüket is, mely természetesen messzemenően támogatja a ZIP-pelést a PKZIP és PKUNZIP, illetve PKZMENU néven elterjedt tömörítő/kibontó programokkal.
A STUPENDOS egy jól sikerült, felhasználóbarát lemezmenedzser utility. Állományok megjelenítése és manipulálása mellett különböző rendezettségben listázza ki az állományokat, megmutatja a könyvtárak fastruktúráját, lemezzel, memóriával kapcsolatos információkat bocsát rendelkezésünkre, csak adott feltételeknek megfelelő állományok listáját is kérhetjük, mozgathatunk állományokat könyvtárak között, mindezt egérrel is vezérelhetjük. Másoláskor, ha nincs elég szabad lemezterület, nem hagyja abba: a rövidebb állományokat, amelyek elférnek, bemásolja.
A lemez második csomagja a PKUNZIP kiváltására készült PKZMENU . Ez egy teljesen menüvezérelt (egérrel is kezelhető) kibontó program .ZIP kiterjesztésű tömörített állományokhoz. A lemezen a részletes (>200 K) angol nyelvű leírás mellett magyar ismertető is található.
Betekinthetünk a .ZIP állományok tartalmába, technikai információt kapunk a tömörített állományokról, a megjelenítendő állományokat több szempont szerint rendezi, jelszóval védett állományok is kicsomagolhatók, programból kezelhetők a DOS-könyvtárak. VGA 50 soros és EGA 43 soros módot is ismer, egérrel is vezérelhető, hálózatban, laptop gépeken is működik, DESQView környezetben is.

**Lemezszám:** 513

**Név:** **Edna's Cook Book**

**Szerző:** Specialty Microware, USA, 1987.

**Leírás:** Szakácskönyvkészítő program.
A program segítségével minden háziasszony vagy szakácskodásra hajlamos férfiú elkészítheti saját receptgyűjteményét, tetszőleges összeállításban. A bevitt receptek nyomtathatók, újra szerkeszthetők, listázhatók, titkosíthatók, vagy egy szöveges állományban tárolhatók. Ezt egy megfelelő szövegvagy kiadványszerkesztő programmal valódi könyv formára is átalakíthatjuk.

**Lemezszám:** 514

**Név:** **Paint Shop Pro**

**Szerző:** JASC Inc. USA, 1991.

**Leírás:** Grafikus program (Windows alá).
Ez egy olyan Windows program, amely TIFF, GIF, TGA, WPG, BMP, PCX, MAC, MSP, IMG, PIC, RLE, DIB és JAS formátumú állományokat kezel. Megjeleníti, konvertálja, módosítja vagy kinyomtatja ezeket. A módosítás fogalmába beletartozik az átméretezés, „trimming", szűrők állítása, palettamanipuláció és még sok egyéb. Képernyőlementésre is alkalmas. Megjelenítéskor a kinagyítást (zoom) is lehetővé teszi. Módosítás esetén el is forgathatunk.
Mielőtt a Paint Shop Próval dolgoznánk, Windowsból le kell futtatnunk a Setup programot. Nem elég tehát a tömörített állományt a lemezre bemásolnunk, kicsomagolnunk. Utána még ez a lépés is szükséges.

**Lemezszám:** 515

**Név:** **Graphic Workshop 6.1**

**Szerző:** Alchemy Mindworks Inc., USA, 1991.

**Leírás:** Grafikus állományok kezelése.
A GRAPHIC WORKSHOP egy könnyen kezelhető, menüvezérelt program, amely grafikus állományok kezelésére alkalmas. Csaknem minden elterjedt formátumot ismer: Macpaint, GEM/IMG, PC Paintbrush PCX, CompuServe GIF, TIFF, WordPerfect Graphics WPG, Deluxe Paint/Amiga IFF/LBM, PC Paint Pictor PIC, Truevision Targa, Windows 3 BMP, Microsoft Paint MSP, Encapsulated PostScript EPS, önmegjelenítő EXE képek, Halo CUT, szöveges állományok, 24 bites fájlok.
Az alábbi műveleteket végezhetjük ezeken az állományokon:
— Belenézhetünk.
— Konvertálhatunk két formátum között (némi megszorítással).
— Kinyomtathatjuk az állományokat LaserJet Plus kompatibilis vagy PostScript lézernyomtatóval, mátrixnyomtatóval. Színes képeket is nyomtathatunk vele színes PostScript nyomtatóra vagy tintasugaras nyomtatóra.
— Átalakíthatjuk a színest fekete-fehérré.
— Megfordíthatjuk.
— Forgathatjuk.
— Skálázhatjuk.
— Csökkenthetjük a bennük levő színek számát.
— Élesíthetünk, tompíthatunk.
— Kisebb állományokat készíthetünk belőlük.
— Teljesen új állományokat scannelhetünk be, feltéve, hogy olyan scannerünk van, amelyet a rendszer támogat.
— Színes állományok színkiegyenlítése, fényességállítás.

Tetszőleges méretű állományokkal dolgozhatunk. Ha nem talál elegendő memóriát, lemezre dolgozik.
**Konfiguráció:** Az összes elterjedt grafikus kártyát támogatja: Hercules, CGA, EGA, VGA. Legalább DOS 3.3 verziót igényel.

**Lemezszám:** 516

## Név: BLASTER Master 4.8

**Szerző:** Gary Maddox, USA, 1991.

**Leírás:** Grafikus zenei editor.
A BLASTER Master Sound Blaster típusú .VOC, .WAV, .SND kiterjesztésű állományokkal dolgozik. A regisztrálatlan shareware verzióval csak 25 másodperces hangállományok kezelhetők, a regisztrált verziónál nincs ilyen korlát.
Maga a program egy igen gyors EGA/VGA grafikus editor, amellyel a hangállományokból a szükségtelen mintákat eltávolíthatjuk (elejéről, végéről; kiválaszthatunk részeket, ezekből új állományt képezhetünk. A kivágási pontok pontos kiválasztásakor nagy segítséget jelent, hogy a teljes minta grafikusan jelenik meg a képenyőn, másodpercenkénti osztásban. A feldolgozáshoz legtisztább pontok kiválasztása a 0 átmenetek alapján történhet. A képernyő színei funkciós billentyűkkel (F1 — F5) változtathatók. Visszajátszáskor pulzáló jel mutatja, hogy hol tartunk a dallamban. A digitális feldolgozási folyamatok aktivizálása egérrel történik, az esetek többségében az Alt billentyű és a funkció első betűje is ugyanezt az eredményt adja.
A BLASTER Master hangkártya nélkül is működik, de természetesen ekkor a PLAY, RECORD és a SCOPE műveletek értelmetlenek.
Konfiguráció: EGA/VGA grafikus kártya, egér, merevlemez, lehetőleg gyors PC.

**Lemezszám:** 517

## Név: SoundFX-Blast

**Szerző:** L.J. Foletta, Silicon Shack, USA, 1991.

**Leírás:** Grafikus editor SoundBlasterhez.
A SoundFX-Blast segítségével felvehetünk, létrehozhatunk, szerkeszthetünk, módosíthatunk, lejátszhatunk és tárolhatunk hangokat. Semmiféle korlátozás nincs ezekkel kapcsolatban. Vezérelhető egérrel vagy billentyűzetről.
Hangok létrehozása a generátorműveletekkel történik. Zajokat, egyszerű dallamokat, összetett hangzásokat, FM- hangokat tudunk így kialakítani.
Generáláskor az aktív hangállományt írjuk felül a hullámforma bal oldalán kijelölt ponttól kezdődően a jobb oldali kijelölésig.
Zajgenerátor — a hullámforma kijelölt részét véletlenszerű fehér vagy nagy sávszélességű zajokkal írja felül. Az amplitúdót mi állítjuk be. A „Filter" művelettel más típusú zajokat is előállíthatunk.
Tiszta szinuszhullámokat, négyszög- és háromszögjeleket hoz létre a hanggenerátor, 1 Hz és 20 KHz közötti frekvenciával, beállítható amplitúdóval, az ismétlődési ciklus 1 és 99% között lehet.
Az AM-generátorral amplitúdómodulált szinuszhullámokat állíthatunk elő, megadhatjuk a vivőfrekvenciát, az amplitúdót, a modulálás százalékát.
Az FM-generátorral bonyolultabb hangzásokat alakíthatunk ki a frekvenciamodulált szinuszhullámok felhasználásával. Beállíthatjuk a vivőfrekvenciát, az amplitúdót, az eltérési tényezőt. Így a szirénahangtól a géppisztolykattogásig a legkülönbözőbb hangeffektusok érhetők el.
Amatőr műhelyben akár ez a demováltozat is alkalmas drága műszerek helyettesítésére (szinusz-, háromszög-, négyszöggenerátor). Blokkos üzemmódja alapján Wobbler generátor imitálására is alkalmas. (Különböző frekvenciájú blokkokat tudunk egymás mögé tenni.)
Saját SoundFX-Blast állományok mellett bármely más szoftver által létrehozott hangállományt kezel, feltéve, hogy a 8 bites minták ASCII formátumúak.
A lemezen található PLAYSB.EXE program nem része az editornak, DOS szintről indítva önállóan játssza le a SoundFX-Blast hangállományokat.
A SoundFX-Blast maximális helyigénye DOS 5.0 esetén 350 K. Ez 27 és 82 másodperc közötti felvételnek felel meg.
Konfigurációs igény: IBM PC, 512K memória, matematikai koprocesszor ajánlott, CGA, EGA, MCG, VGA vagy HGA , MS-DOS 2.10 vagy magasabb verzió, SoundBlaster kártya, mikrofon, kábel.

**Lemezszám:** 518

## Név: Audiostar 1.0

**Szerző:** Lars Schenk & Frank Horn, Németország, 1991.

**Leírás:** Grafikus zenei editor.
Az AudioStar mind SoundBlaster (VOC), mind pedig VoiceMaker II (ADS) hanganyag feldolgozására alkalmas. A jövőben AdLib és VoiceMaker I (SND) irányba kívánják továbbfejleszteni, így ezen mintákat és állományokat is kezelni fogja.
A könnyebb áttekinthetőséget szolgálja a grafikus környezet, amelybe az AudioStar digitalizált mintákat betölthetjük és feldolgozhatjuk. A minták egy ablakban grafikus görbeként jelennek meg. Ezeken könnyen észrevesszük a szüneteket, a hangosabb, halkabb részleteket, így nem szükséges hosszas fejtörés az esetleges módosításokhoz.
A rendszer szolgáltatásai:
— hangminták feldolgozása;
— SoundBlaster és VoiceMaker II támogatás;
— grafikus környezet redőnymenükkel;
— VGA, SVGA, MCGA, EGA és Hercules grafikus kártyák támogatása;
— a leggyakoribb műveletekhez „hotkey";
— mind egérrel, mind billentyűzetről vezérelhető;
— különböző minták felvétele és lejátszása;
— sokrétű blokkfeldolgozási lehetőség;
— minták keverése;
— különböző visszajátszási lehetőségek;
— blokkok betöltése és tárolása;
— visszhangosítás;
— blokkok átfordítása;
— hangerő-változtatás;
— kivágás/beillesztés;
— belső hangszóró vagy Voicemaker II hangszóró használata;
— ADS-VOC és VOS-ADS átalakítás (így VOC állományok is lejátszhatók a PC belső hangszóróján);
— Audiomaster II (Amiga IFF) állományok is betölthetők.
Szükséges konfiguráció:
IBM PC min. 512 K RAM-mal; DOS 3.0 vagy magasabb verzió; Hercules, CGA, EGA, VGA, MCGA vagy Super VGA grafikus kártya.
Ajánlott:
— Microsoft-kompatibilis egér;
— 640 K munkamemória;
— merevlemez;
— SoundBlaster kártya vagy VoiceMaker II.

**Lemezszám:** 519

## Név: Big 2

**Leírás:** Kártyajáték.
A BIG2 játék Hongkongból származik, ahol a szerencsejátékok a mindennapi élet szerves részét képezik. A keleti népesség csaknem 99%-a ismeri, ez a második legnépszerűbb játék, csak a nálunk is elterjedt Mah-jong előzi meg. Szabályai igen egyszerűek, ennek ellenére a vérbeli játékosok szinte csodákat művelnek. A cél az, hogy elsőként szabaduljunk meg összes kártyánktól, miközben minél több lapot viszünk ütéseinkkel, a többi játékos kezében pedig minél több lap maradjon. A póker és a bridzs elemeit fedezhetjük fel benne.
A nálunk megszokottól eltérő a színek és számok sorrendje: leggyengébb szín a káró(!), ezt követi a treff, kőr, pikk. Legkisebb szám a 3, ezt követi 4,..,10,B,D,K,Á, legnagyobb lap a 2-es!
Konfiguráció:
A játékhoz EGA vagy VGA grafikus kártya szükséges, és persze egér is.

**Lemezszám:** 520

## Név: LP 2.1

**Szerző:** Armin Winkler, Németország, 1991.

**Leírás:** Hanglemez-katalogizáló.
Menüvezérelt lemezkatalogizáló rendszer. Hanglemezenként 9 képernyőnyi információt tárolhatunk, egy lemezhez

ALAPLAP
CSOMAGKÜLDŐ SZOLGÁLAT
POSTA

Jelenleg mintegy 1000 szoftverből, illetve külföldi szakkönyvből válogathat. Az aktualizált lista megtalálható mostani számunk lemezmellékletén. A megrendelt szoftvert vagy külföldi szakkönyvet postai utánvétellel 2 héten belül házhoz szállítjuk.

**MEGRENDELÉS**

Megrendelem postai utánvétellel az alábbi termékeket. A vételárat és a postaköltséget átvételekor kifizetem.

A) SZOFTVEREK:

..........................................................

..........................................................

..........................................................

B) SZAKKÖNYVEK:

..........................................................

..........................................................

..........................................................

Dátum:

(aláírás)

# Amíg a készlet tart...

Megrendelem az Alaplap kiadványsorozataiban (Könyvek, Füzetek, Lemezek) eddig megjelent (és még kapható) művek közül az alábbiakat:

## ALAPLAP KÖNYVEK

... pld: Nagy Gábor: Tömör gyönyör, 256,– Ft

... pld: Kis János–Szegedi Imre: Új víruslélektan, 256,– Ft

... pld: Kis János–Szegedi Imre: Vírushatározó, 256,– Ft

## ALAPLAP FÜZETEK

... pld: Detrik Péter: Az SQL nyelvről, 375,– Ft

## ALAPLAP LEMEZEK

... pld: Bliss főkönyvi program demója és leírása, 750,– Ft

... pld: Norton Guide keretprogram leírása, 500,– Ft

... pld: PathMinder segédprogram leírása, 500,– Ft

... pld: CSProlog nyelv leírása, 1000,– Ft

... pld: LIM EMS 4.0 memóriakezelő leírása, 1000,– Ft

... pld: Windows 3.0-hoz magyar betűkészletek, 1000,– Ft

# PC Turbo Klub

## VÁLASZLAP

Eddigi PC Turbo Klub-tagoknak nem kell kitölteniük és beküldeniük, ha címüket az új címadatbázisban is szerepeltetni akarják. (Lásd erről részletesebben cikkünket az Alaplap 92/8. számának 54. oldalán.)

☐ A PC Turbo Klub tagja vagyok, de kérem, hogy az új címadatbázisban címem már ne szerepeljen.

☐ Nem vagyok a PC Turbo Klub tagja, de szeretném, ha túloldalt megadott címemet az új címadatbázisba felvennék.

Tudomásul veszem, hogy a PC Turbo Klub új címadatbázisába való felvétel esetén címemet az Alaplaptól szakmai címanyagot kérő cégek és intézmények direct mail akcióikhoz (meghívók, prospektusok stb. kiküldéséhez) felhasználhatják.

Dátum:

..........................................
aláírás

# INFORMÁCIÓKÉRÉS

Kérem, hogy az itt általam **BEKARIKÁZOTT KÓDSZÁMÚ** hirdetésekkel kapcsolatban küldjenek részemre bővebb tájékoztatást.

Beküldhető: 1992. október 31-ig

| | | | | |
|---|---|---|---|---|
| 01 | 02 | 03 | 04 | 05 |
| 06 | 07 | 08 | 09 | 10 |
| 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 |
| 31 | 32 | 33 | 34 | 35 |
| 36 | 37 | 38 | 39 | 40 |
| 41 | 42 | 43 | 44 | 45 |
| 46 | 47 | 48 | 49 | 50 |
| 51 | 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 | 60 |
| 61 | 62 | 63 | 64 | 65 |
| 66 | 67 | 68 | 69 | 70 |
| 71 | 72 | 73 | 74 | 75 |
| 76 | 77 | 78 | 79 | 80 |

ALAPLAP
1992/9
SZEPTEMBER

## FELADÓ:

**A) Egyéni érdeklődő:**

**Név:** ................................................

**Utca, házszám:** ................................................

**Helység:** ................................................

**Irányítószám:** ................................................

**B) Vállalati érdeklődő:**

**Cégnév:** ................................................

**Ügyintéző:** ................................................

**Utca, házszám:** ................................................

**Helység:** ................................................

**Irányítószám:** ................................................

**Telefon/Fax:** ................................................

Belföldön díjmentesen feladható

Cédrus Kiadó
Pf. 74

Budapest

1441

---

FELADÓ:

Név: ................................................

Cég: ................................................

Utca, házszám: ................................................

Helység: ................................................

Irányítószám: ................................................

Telefon/Fax: ................................................

Belföldön díjmentesen feladható

Cédrus Kiadó
Pf. 74

Budapest

1441

---

**FELADÓ:**

Név: ................................................

................................................

Foglalkozás/Beosztás: ................................................

Cím: ................................................

................................................

Helység: ................................................

Irányítószám: ................................................

Belföldön díjmentesen feladható

Cédrus Kiadó
Pf. 74

Budapest

1441

---

FELADÓ:

Név: ................................................

Cég: ................................................

Utca, házszám: ................................................

Helység: ................................................

Irányítószám: ................................................

Telefon/Fax: ................................................

Belföldön díjmentesen feladható

Cédrus Kiadó
Pf. 74

Budapest

1441

# A LEMEZMELLÉKLET TARTALMA:

- Poli(p)technika matematikából
- SolarSoft lemezkóstoló
- A többfeladatos működés szimulációja
- Ismerkedés a Snobol nyelvvel
- Nagy szövegállományok szétdarabolása
- Segédprogram a CEB-vírusok ellen
- Szorzótábla-gyakorlat kisiskolásoknak
- Játék a szavakkal
- Gumidominó
- Alaplap Posta — szoftverek, szakkönyvek kínálata



16 dal tartozhat. Ezer lemez nyilvántartására alkalmas. Lemezenként felvihetjük az előadók nevét, a dalok címét, időtartamát, saját értékelésünket (menüből választható minősítések), kategóriát (szintén menüből választunk: rock, komolyzene, country...). Különböző rendezettségben kérdezhetjük le az állományt: lemezsorszám, előadó, kategória, minősítés... Kereséskor tetszőleges szűrőt is állíthatunk az állományra. Címkéket is nyomtathatunk külön az egyes lemezekhez, vagy az összes lemezhez egyszerre. Statisztikák százalékosan és grafikusan is megjeleníthetők, akár az értékelés, akár a kategóriaazonosító alapján.

**Lemezszám:** 521

## Név: Label Magic 3.0

**Szerző:** Joseph M. Albanese, USA, 1992.

**Leírás:** Grafikus címkegenerátor.
Levelezésre is alkalmas grafikus címkéket készíthetünk 2500 pontos ikonokkal, mivel saját beépített ikoneditort is tartalmaz.
PRINTSHOP és PRINTMASTER ikonok átalakítkatók Label Magic ikonokká.
Körleveleket készíthetünk (batch nyomtatás!) grafikus ikonokkal vagy azok nélkül is.
Közvetlenül a borítékokra nyomtathatjuk az ikonokat, címeket. Természetesen csakis akkor, ha rendelkezünk olyan nyomtatóval (például: HP DeskJet, IBM Proprinter), amelybe be tudunk fűzni borítékot.
Ugyanazokat az ikonokat és/vagy címeket nyomtathatjuk levélfejlécként, amelyek a címkékre kerülhetnek.
A programcsomag részeként szállított CONVICON.EXE segédprogram a Label Magic ikonállományokat (.ICN kiterjesztésűek) konvertálja különböző formátumokba, más formátumokat pedig .ICN alakba. Többek között az ASCII formátumot is kezeli. Ezek a szöveges ASCII ikonok a továbbiakban szöveges állományokban is felhasználhatók.
A Label Magic 2500 pontos ikonjait Pascal programokban is használhatjuk a CONVICON átalakítás után.
Nyomtathatunk Label Magickel LQ-2500, Panasonic KX-P1124 és ezekkel a típusokkal kompatibilis 24 tűs nyomtatóra.
**Konfiguráció:** CGA, Hercules (Hercules esetében nincs egértámogatás), MCGA (Model 30), EGA, AT&T 640x400 HIRES, vagy VGA (színes és monokróm).

**Lemezszám:** 522

## Név: Galactic

**Szerző:** Brian Goble, illetve Charles A. Hardy

**Leírás:** 2 játék.
A GALACTIC könyvtár a tömörítetlen Galactic Battle játékot tartalmazza.
Egy űrbeli lövöldözős játékról van szó, a játékos egy űrhajóban ül, ő a kapitány. A gonosz földönkívüliek támadásait kell kivédenie. Cél: minél több ellenfelet megsemmisíteni, miközben az űrhajónak egyre jobbak lesznek a képességei.
Számtalan szintje van a játéknak, az egyes szinteken 11 különböző ellenséges horda támadására kell számítanunk. Ha mind a 11 támadást kivédtük, eggyel feljebb kerülünk, ami természetesen mindig nehezebb. Az ellenségek pedig mindig hullámokban jönnek, nem egyedül, hanem tízesével.
Érdemrendeket is szerezhetünk, ezeket védelmi és támadó eszközök vásárlására használhatjuk fel.
A játék vezérelhető joystickkal, egérrel vagy billentyűzetről.
A játék CGA (4 szín), EGA, VGA (16 szín) grafikus kártyákhoz készült. Felbontása 320 x 200 képpont. Herculesen is játszható CGA-szimulációval.
A MINEF115 könyvtár MINEF115.exe állománya önkicsomagoló.
A MINEFIELD 1.15 csak VGA kártyával és egérrel működik. 350 K szabad lemezterületre és 400 K memóriára van szüksége. Csak merevlemezről érdemes indítani, nagyon lassú lenne máskülönben.
A feladat egyértelmű: fel kell szednünk az összes aknákat. Tudjuk, hány akna van összesen, rendelkezünk aknakeresővel. Az aknakereső megmutatja, hány akna van a környező nyolc kockában. Ahhoz, hogy használhassuk az aknakeresőt, rá kell lépnünk egy kockára, itt fontos az óvatosság — és a szerencse. A bal oldali egérgombbal akkor kattintunk, amikor olyan kockára lépünk, amely alatt szerintünk nincs akna. A jobb oldalival pedig akkor, amikor kijelölünk egy gyanús kockát, amely alatt aknát sejtünk.
15 pálya közül választhatunk. A zene is ki/be kapcsolható. Pályánként beállíthatjuk, hogy hány akna legyen (min. 10, max. az összes kockák száma), természetesen mindenhol van alapértelmezés is.

**Lemezszám:** 523

## Név: Facetris

**Leírás:** Arckirakó (Tetris-szerű).
Először volt a Tetris, ezt követte a Welltris, most a sorozat harmadik tagjával, a Facetris-szel játszhatunk. A játék menete hasonló az előző kettőéhez. Itt arcrészletek jönnek lefele, a megfelelő elemeket kell összeválogatnunk. Magától értetődő, hogy az áll fölé a száj jön, majd az orr...
A játék 60 fejet tartalmaz, 10 nehézségi szint közül választhatunk. A legjobb 10 játékos kerül fel a dicsőségtáblára. Játszhatunk haladó módban is. A demóval maximum 1500 pontig mehetünk.
Vezérlése: vagy kurzorbillentyűkkel, vagy betűkkel.
**Konfiguráció:** Hercules, CGA, EGA, VGA vagy SVGA grafika.

**Lemezszám:** 524

## Név: Ford Simulator (2 lemezen)

**Leírás:** A 2 lemez rengeteg információt tartalmaz a Ford cég termékeiről, ugyanakkor jól sikerült autószimulátor is.
A főmenüből a Driving Simulatort választva beállíthatjuk a vezérlést: billentyűzetről, egérrel vagy joystick-kel; a hangot ki/be kapcsolhatjuk; három nehézségi szinten, különböző pályákon vezethetünk. A tesztpálya alakja háromféle lehet, mindegyiken 1, 2 vagy 3 kört tehetünk meg, a városi pálya repülőteret, bevásárlóközpontot vagy belvárosi forgalmat jelenthet; 5, 10 vagy 15 mérföldes óránkénti sebességgel haladhatunk a terelőutakon.
A Bemutatóterem menüpontban a kiválasztott Ford-termékek megjelennek a képernyőn, további részletezett adatokat kaphatunk mindegyikről.
A Vásárlói Tájékoztatóból megtudhatjuk az egyes típusok alapfelszereltségét, a beépíthető extrákat.
Magyaroknak természetesen nem igazán szükséges az Infocenter menüpont, az amerikai érdeklődők azonban bizonyosan örömmel vették, hogy még az egyes típusok után fizetendő havi részletet is kiszámolták helyettük.
Konfiguráció: CGA, EGA, VGA grafikával működik.

**Lemezszám:** 525

## Név: VGA-Copy 4.1

**Szerző:** Th. Mönkeheimer, Németország.

**Leírás:** Grafikus floppymásoló.
Az MS-DOS környezetben használatos lemezformátumok másolására készült a program. Ezenfelül nagyobb szektorszámú, különleges formátumú lemezeket, DOS-idegen bootszektorokat is felismer, kezel. Példa erre az Atari ST. Az eltérő DOS-formátumokat tetszőlegesen konvertálja, olyan formátumokat is képes előállítani, amelyeket a DOS nem lenne hajlandó kezelni. Részletezve: 1,44 MB, 5,25"; 720 KB, 5,25"; 360 KB, 3,5" és 1,2 MB, 3,5".
A legérdekesebb ezek közül az első típus, a többit más programok is kezelik. A 80 sávnál nagyobb (pl. 1,72 MB) formátumokat is felismeri, és helyesen másolja.
Nem teljesen automatikusan ugyan, de egyszeri megadás után már önállóan megtalálja a VGA-Copy a vírusfelismerők/irtók és egyes DOS-adatok tárolóhelyét.
A fejlesztők felkészítették a VGA-Copy-t John McAfee antivírusprogramjainak használatára. Ezeket a vírusölőket nekünk magunknak kell beszereznünk, és a gépre feltennünk.
**Konfiguráció:** VGA-grafika.

**Lemezszám:** 526

**Név:** UIH

**Leírás:** 2 lemez.
UIH = Utilities im Hintergrund, magyarul: segédprogramok a háttérben.
Maga az UIH két főprogramból áll, amelyek rezidensként vannak betöltve. Szerves részét képezi az UIHSHELL menürendszer is.
UIH.EXE — rezidensként mindössze 4,5 K memóriát foglal!
— Memóriával kapcsolatos információk:
Memóriakiosztás (DOS/EMS/XMS); programok helyfoglalása; interruptkiosztás; az összes CMOS/rendszer-RAM tartalmak; megnyitott állományok; perifériálista;
— Zsebszámológép
— Asztali számológép
— Programozható számológép
— Tudományos számológép
— Formula interpreter
— A kiszámított értékeket átadhatjuk az előtérben dolgozó programnak.
— Óra: nagyméretű digitális óra megjelenítése; rendszeridő beállítása; rendszerdátum beállítása.
— Határidőnapló:
— — 1-4 negyedévet kezel a képernyőn
— — Infoablak bevillantható
— — Tevékenység nyilvántartás (ToDo) automatikus ismétlési lehetőséggel (1,2 hét — 1,2,3,6 hónap)
— — Ünnepnapok
— — Határidő automatikus jelzése
— — Határidős tevékenységek kezdő dátumainak nyilvántartása, max. 99 napos előrejelzés
— — Naptári hét megjeleníthető!!
— — Naptár aktualizálható
— — Határidő lista nyomtatható (egy hónapra)
— — Éves határidők, tevékenységlista
— — Felvilágosítás a határidőkről
— — Határidőnként egy szöveges mező (64K)
— — Hálózati lehetőség: más állomások határidős tevékenységeihez is hozzáférhetünk. Hozzáférés-védelem: személyes határidőkre, illetve tevékenységekre
— — Fix ablakban görgethetők a havi bejegyzések
— ASCII-táblák: mind a 256 ASCII-jellel; csak vonalas grafikával; csak vezérlőjelekkel; jelek átvihetők az előtérben dolgozó programba.
— Tasztatúra: billentyűzet megjelenítése és ellenőrzése; tasztatúrakód megjelenítése.
— Jegyzetfüzet: 10 jegyzetlap dolgozható fel gyorsan, anélkül, hogy állománynevet kellene rendelnünk hozzájuk (szövegszerkesztőhöz hasonló műveletekkel).
— Editor
— — Gyors Wordstar-kompatibilis editor
— — Redőnymenü-vezérlés
— — Szótagolás
— — Blokkműveletek az eltolás, másolás, betöltés, írás stb. utasításoknál
— — Clipboard
— Makró billentyűkiosztás: tetszőleges billentyűsorozatot rendelhetünk egy adott billentyűhöz, illetve billentyűkombinációhoz (pl. <F12> vagy <ALT><Z>); a makrók állományban is tárolhatók.
— Állománykezelés
— — Két állománymegjelenítő ablak és két könyvtárablak lehet egymás mellett.
— — Csaknem minden művelet végrehajtható egyetlen billentyű leütésével.
— — Állományok, ill. teljes könyvtárszerkezetek másolása, törlése, betöltése.
— — Állományok, ill. könyvtárak eltolása — akár más meghajtóra is.
— — Állományok átnevezése.
— — Attribútumok beállítása.
— — Állományok keresése (dátum, méret, tetszőleges szűrővel, archív és önkicsomagoló állományokban is keres). A keresőablak utasításai: tartalom; kijelölés és törlés; azonos állományok kiválasztása; rendezés (név, kiterjesztés, dátum, méret szerint).
— — Állományok és könyvtárak gyors keresése.
— — Tetszőleges szempont szerinti rendezettségben jeleníthetünk meg.
— — Lehetséges attributumok és '*' (joker) szerinti kiválasztás is.
— — Állományok összehasonlítása.
— — Állománytartalom bemutatása: szöveges és hexa módban (szövegrészek keresése, blokkírás, blokknyomtatás); dBase állományok; archív állományok (ARC/LHARC/LHA/ZIP/ZOO/ARJ és önkicsomagoló állományok) további feldolgozása is lehetséges; kijelölés és kicsomagolás; állomány tartalma; rekurzív tartalom abban az esetben, ha az állomány ismét archív állomány.
— — Grafikus állományok (Az ALCHEMY program által támogatott formátumok, például PCX/GIF/JPG/BMP; ALCHEMY.EXE-re is szükségünk van.)
— Programok futtatása: közvetlenül (.BAT, .EXE, .COM); állománycsoportok (pl.: .TXT, .DOC WORD-del, .BAS QBASIC-kel, stb.); programmenü.
— DOS-parancsok végrehajtása (pl. DIR C:*.*)
— Tömörítőprogramok támogatása (állományok ki/be csomagolása ismert tömörítőkkel, mint pl. ARJ, PKZIP).
— EXE/COM tömörítéstámogatás PKLITE-tal. (.EXE und .COM állományok tömörítése.)
— Kisebb állományokhoz editor
— Könyvtárak létrehozása
— Könyvtárak átnevezése
— Meghajtóáttekintés (Kapacitások grafikus megjelenítése)
— Floppyformattálás (Adat-/Bootlemez)
— Floppy törlése (Nagyon gyors)
— Könyvtár ágak ki/be villantása
— DBase-kompatibilis adatbázis
— — Kiválasztások (ÉS/VAGY/NAGYOBB/... kapcsolatok mezőnként)
— — Gyors keresés (INDEX)
— — Tetszőleges keresés (Minden mező, ill. rekord)
— — Többszörös módosítás
— — Újraszervezés
— — Listanyomtatás
— — Címkenyomtatás
— — Körlevelek
— — Memómezők (szöveges)
— — Maszkfelépítés/ Browse felépítés és listaformátum beállítható.
— — Adatrekordok kijelölése pl. címkenyomtatáshoz vagy körlevélhez.
— További lehetőségek
— — Hálózati használat (pillanatnyilag max. 100 felhasználó)
— — Az egyes felhasználókhoz saját paraméterállományt rendel (színek/más beállítások...)
— — Rezidens DOS-Prompt. Csaknem bármely programból bármely időpontban gombnyomásra indulnak a DOS utasítások.
— — UIH meghívása grafikus programokból is lehetséges
— — VGA (262144 színből 16) támogatás. A paletta UIH-on belül bármikor változtatható <SHIFT><F5> -tel.
— — EGA/VGA támogatás, EGA 25/43 soros mód, VGA 25/28/30/34/43/50/60 soros mód. Az UIH csaknem minden fontosabb műveletéhez eltérő sor módokat rendelhetünk.
Pl.: Állománykezelő 34 soros
Editor 60 soros
Határidőnapló 43 soros
Adatbázis 30 soros
— Az UIH bármely állapotában megjeleníthető a képernyőn az előtérben dolgozó program <ALT><F5> leütésével (csak szöveges módban)
— Bármely művelethez (Állománykezelés, editor stb.) rendelhetünk „hotkey"-t.
— Egértámogatás
— Online segédszöveg
— Az előtérbeli képernyő szövegrészei átvehetők akár input mezőbe, akár az editorba.
UIHDOS.COM — kibővített sor editor DOS szinten. Rezidensként installálva F1 leütésekor az összes utasítással kapcsolatos információkat megkaphatjuk.
UIHSHELL.EXE — menürendszer, integrált DOS prompttal. Segítségével csoportokba, illetve alcsoportokba jegyezhetünk be programokat. Paramétereket rendelhetünk a programokhoz, a lekérdezés felhasználóbarát.
— Módosítható ablak (méret/pozíció).
— VGA esetén 28/30/34/43/50/60 sor állítható be.
— Egértámogatás.
Az integrált DOS promptot SAA szerint illesztették.
— SAA-kezelés.
— Utasításnaplózás.
— Utasításeditor.
— Programok hívása jokerrel (*,+,!).
— Memóriamegjelenítés.
— Integrált editor AUTOEXEC.BAT és CONFIG.SYS-hez.
— Batch editor..
— Jelszókezelés, védelem.
**Konfiguráció:** legalább 1,2 M szabad lemezterület szükséges.

## Itt a pénz, hol a pénz?

# Fogd a mágneskártyád, és légy vidám!

**Vajon az eddig otthon (zárt ajtók mögött) számolgatott bankók(ötegek) birtoklásának örömét miként helyettesít(het)i egy műanyagból készült dolog, hogyan lehet ezt használni, és vajon mire, és mikor, és ugyan miért jó ez? Milyen szerepe van ebben az egészben a számítógépnek?**

Míg az emberek szabadságukról hazatérve egyre hűvösödő estéken idézik fel az elmúlt nyár élményeit, addig a bankok egyes pénzkezelésekkel kapcsolatos tevékenysége körül alakul majd kánikulai hőség. Teszem ezt a kijelentést annak okán, hogy az eddigi pénzkezelési hagyományokkal teljesen ellentétes pénzkezelés körül magam is bábáskodtam. (A jövevény — Kelet-Európa első online rendszere — több mint fél éve kifogástalanul üzemel.) A változás a készpénzkímélő fizetőeszközök mind szélesebb körű elterjedésében teljesedik ki. Nem szándékom ennek különböző előnyeit ecsetelgetni, ezt valószínűleg megteszik majd minden percben az erre szakosodott médiumok, de a számítástechnikai megoldásokról tőlük nem fogunk hallani. E cikk viszont ezekről szól.

### Megy a forint vándorútra

Mi indokolja a készpénzkímélő fizetőeszközök alkalmazását? Vegyünk egy egyszerű példát.

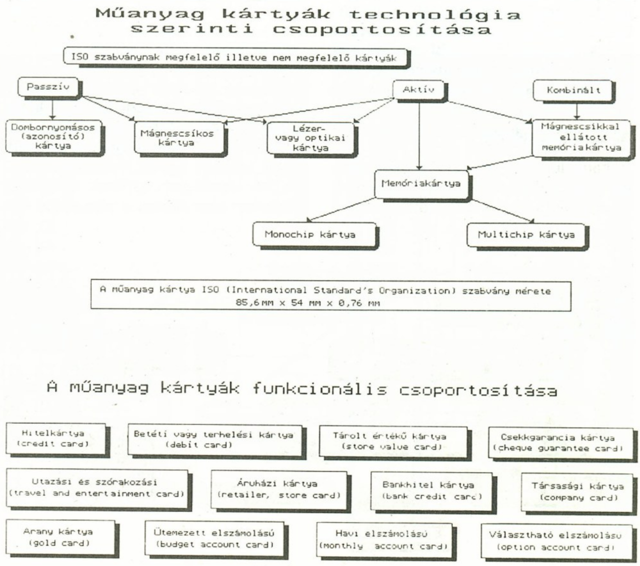
Egy vállalat havonta sok dolgozójának munkabérként pénzjegyeket ad. A vállalat a dolgozók bérének számfejtését követően címletjegyzéket készít, mely címletjegyzék tartalmazza az egyénenkénti bérkifizetéshez szükséges bankjegyek darabszámát.

Ezt aztán összesíti és elküldi a banknak. Ott kikészítik vállalatonként a szükséges pénzcímleteket, melyeket a vállalat biztonsági kíséret igénybevételével a kifizetési helyre szállít, majd megfelelő védelem mellett a bankjegykötegeket személyek szerint borítékolják. Ha több kifizetőhely is van, akkor eszerint szétosztják, és a fizetési napon a dolgozók megkapják.

A dolgozók — bármily kevés is a fizetésük — ennek egy hányadát visszaviszik a bankba, ahol a pénzeket szortírozzák, csoportosítják, és visszateszik a trezorba. A bankók egy része az üzletekből fordul vissza a bankba: az üzletek összesítik a forgalmukat, és az összegyűlt bankjegyeket beviszik a bankba — ugyan ki vállalja a pénz őrzésének tetemes költségét, kockázatát? —, ahol azokat a már említett trezorba helyezik el. A folyamat sok embernek ad munkát, mely munka kizárólag a pénzkezeléssel kapcsolatos, és nem is kevés.

Van-e lehetőség a tevékenység egyszerűsítésére, és van-e erre példa?

Természetesen elképzelhető a készpénz helyettesítése egy az egyben. Erre a célra bármely eszközt igénybe vehetünk, hiszen a kezdetben vala erre a célra kő, bronzkarika, só, és még sok minden más. Mostanság a kártya járja. (Mellékesen megjegyzem, hogy a kártyákat nem csak fizetőeszközként alkalmazzák, például az egészségügyben jó




*1. ábra*

szolgálatot tett kártyatípusokat is bemutatok — 2. ábra.)

Tisztem szerint nem, de mondandóm előtt mégis a — pénzkiváltásra alkalmazott kártyatípusok közül egy bizonyos felépítésűvel, nevezetesen a mágnescsíkkal ellátott kártyával kapcsolatos — szolgáltatásokról is kell szólnom. (A szolgáltatás igénybevételére betéti kamattal csalogatnak, de azért költségeket vonnak le.)

A mágnescsíkos kártyák kibocsátói a következőket kívánják a kártyahasználók részére biztosítani:

— készpénzfelvétel, melyet 24 órás üzemben működő pénzkezelő automaták tesznek lehetővé;

— bankablaki műveletként ki- és befizetések;

— vásárlás készpénz helyett kártyával.

A rendszer elvi működése a 3. ábrán látható módon írható le.

A készpénzfizetési eszköz kiadására jogosult szervezet — a továbbiakban bank — valamilyen, a célt szolgáló eszközt (jelesül kártyát) bocsát ki a szolgáltatást igénylők részére. A szolgáltatást igénybe vevőkkel erről közös egyezséget (szerződést) kötnek, amelyben szabályozzák az igénybevétel módját, mértékét, a szolgáltatás ellenértékét, valamint személyesítik az igénybevevő által meghatározott személyek részére az új típusú fizetőeszközt. A személyesített eszközzel a jogosultak — továbbiakban kártyahasználók — forgalmat bonyolíthatnak le az előbb említett célokból. A szolgáltatást igényelhetik természetes és jogi személyek is.

## Az egyezség háttere

A bank továbbá szerződést köt mindazokkal, akik készpénz helyett a kártyát fizetési eszközként elfogadják. Ezeket elfogadóknak nevezzük. Az elfogadók felsorolják mindazon általuk üzemeltetett helyiségeket, amelyekben a rendszer érvényesül. (Például elfogadó a Centrum Áruház és az általa felsorolt fiókjai.) Ezen megállapodást elfogadói szerződésben rögzítik. Minden egyes elfogadó azonosító kódot, és minden elfogadói hely egyedi azonosítót kap. Az elfogadói szerződésben továbbá megegyeznek az esetleges kártyatípusokra vonatkozó elfogadói procedurákra (személyiség igazolása, nagy összegeknél engedélyek kérése, stb.).

## Pénzt csak pénz csinálhat

Vezessük be a hálózat fogalmát. A hálózat nem más, mint a kibocsátó által kiadott kártyák elfogadására létrehozott kapcsolatlánc, melynek célja a szolgáltatás maradéktalan biztosítása. A hálózat generátora a kibocsátó.

A kártyahasználói forgalom azonban akkor válik az ügyfél részére kedvezővé, ha topológiai korlátok nélkül és bármikor hozzáférhet a pénzéhez. Ezt a kártyakibocsátó bankok vagy országos fiókhálózatukon keresztül valósítják meg, vagy egymással megegyeznek, hogy az általuk már kialakított rendszerben más kibocsátó kártyaforgalmát is elfogadják.

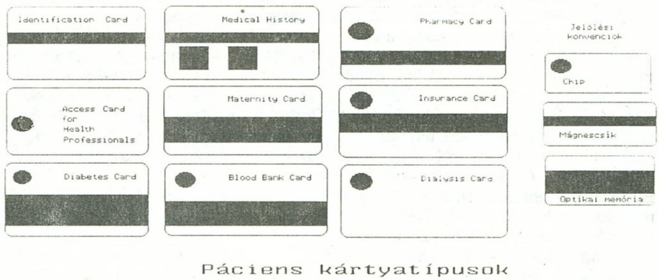
Mindez a következőképpen fogalmazható meg.

A kártyakibocsátók egymás rendelkezésére bocsáthatják hálózatuknak egy vagy több, célszerűen minden végpontját. Ezt a kapcsolatot egy közös megállapodást tartalmazó szerződésben rögzítik.

Az elmondottak alapján a szolgáltatás két síkon realizálódik:

— Lokális (a szolgáltatás kizárólag a kibocsátó hálózatában érvényes).

— Globális (a szolgáltatás az egymással kapcsolatban álló hálózatok minden végpontján igénybe vehető).



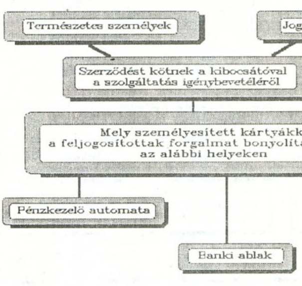

*2. ábra*

A tranzakciók lebonyolíthatóságára, a pénzügyi folyamatok követésére a hálózatoknak és elemeiknek kapcsolatban kell állniuk egymással. A kapcsolattartás módjai:

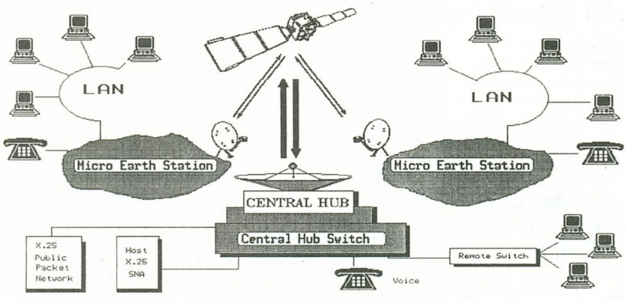
— Verbális kapcsolat. Kézenfekvő megoldás erre a telefon, ha van. Vélhetően nem ideális az ügymenet.

— Formális kapcsolat. A tranzakciókat valamilyen közös megegyezéssel kialakított formátumok alapján (bizonylatok) írják le, és erről tájékoztatják egymást (telefax, telex stb.). Ez a módszer a jelenlegi átutalási gyakorlatba bekapcsolható, számolva az ismert nehézségekkel. Ez sem ideális megoldás.

— Online kapcsolat. A kívánatos tranzakció valós időben a kibocsátónál jelentkezik és könyvelődik. Ez a kapcsolat a legkézenfekvőbb, egyben technikailag a legnagyobb erőforrást igénylő feladat. Megvalósulhat vezetékes úton, rádiófrekvenciás úton és műholdas kapcsolaton.




*3. ábra*




*4. ábra*

## Reményteljes kapcsolattartás

Érezhetőek a telefonhoz jutás nem gyors, de biztató jelei. A gerincvezeték optikai kábelből már az év végére felépül. Legnagyobb szereplői a GIRO francia—magyar együttes, a MATÁV és csapata. Már ezen cikk írásának pillanatában rendszerek üzemelnek, melyek közül a a SIEMENS csomagkapcsolt koncepciójú hálózata igen aktív. Lényege, hogy a rendszerben címzett üzenetcsomagok haladnak a lehető legrövidebb, azaz a legoptimálisabb úton a célállomás felé. Nem pont–pont kapcsolat alakul ki, lefoglalva az átviteli utat, hanem a fizikai közegben egy időben több csomag utazhat, így az átviteli út maximálisan kihasználható. Remélhetőleg ez a szállítási költségekben tükröződni fog.

## Rádiófrekvenciás kapcsolat

Fejlettebb technikával rendelkező országokban célirányos rádiómodemeket alkalmaznak, amivel a gyorsabb adatátvitel mellett kisebb költségek járnak. Ezek közül ígéretes a RAN modemcsalád, mellyel pont–pont-, pont–többpont kapcsolat építhető ki 1,2-128 kbit/s adatsebességig, 928-960 avagy 400-512 vagy 820-960 MHz RF csatornán, 12,5, 25, 100, 200 kHz-es sávszélességgel, 5 W-os adóteljesítménnyel. Megelégedettséggel használják a mezőgazdaság, vegyipar, távközlés, építőipar, oktatás-kutatás, energiaipar és közművek, pénzügyi intézmények, államigazgatás, egészségügy területén — sajnos nem hazánkban.

Általában két év megtérüléssel telepíthetők a rendszerek, ha a frekvenciaengedélyt valaki megkaphatja...

Földgolyónyi méretben online kommunikáció valósítható meg az űrtechnika vívmányaival. Úgy gondolhatjuk, hogy ez a gigantikus méretű, azonnali kapcsolattartás gigantikus költségekbe kerül. Van olyan ajánlatom, melyben 2,5 Mbájt átviteléhez a kapcsolat 250 000 Ft-ba kerül, függetlenül attól. hogy Tokyóról van-e szó vagy Bácsbokodtól. (A 2,5 Mbájt kevésnek tűnhet, gondoljunk azonban az adattömörítésekre!) Hazánkban egy ilyen országos hálózatot az IBUSZ már tartósan üzemeltet. Mivel ez a kapcsolattartás a legelvontabb, a 4. ábrán az ANT Nachrichteintechnik Gmbh által üzemeltetett VSAT Business Satellit Communication System működésének elvi vázlata látható.

Bármilyen információhordozó közeget is választunk, költségminimalizálás miatt megoldandó az információ tömörítése, integritása. Ismertek a népszerű ki-be csomagoló programok, melyek az eredeti adatállományt akár 90%-ban is képesek összecsomagolni.

## Titkos, ami titkos

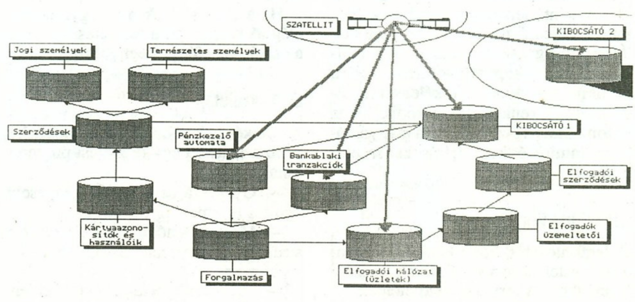
Külön kell szólni azonban az adatok rejtjelezéséről. Míg az inkák a távadatátvitelre, rejtjelezett üzenettovábbításra füstjeleket, a rómaiak dombokra állított szemaforokat, az afrikaiak a tamtamdob hangját használták, ma a haditechnikából átkerült DES titkosító algoritmust használó chipek a járatosak. Az igényesebb érdeklődőnek szívesen figyelmébe ajánlom az evvel a témával részletesebben foglalkozó Nemetz Tibor—Vajda István: Algoritmusos adatvédelem című kiadványt.

A vírusok problematikájához hasonlóan biztosan lesznek mániákus rejtjelezők és ádáz megfejtők. Mindenesetre az efféle védelmen túl a háttértárolókon tartott adatokat csak különböző buktatókon keresztül jutva lehet elérni. Ez a jogosultsági rendszer hierarchiája. A lehető legszélsőségesebb ötletekkel — ujjnyomat egyezése, íriszvizsgálat — igyekeznek az adatokat elszigetelni, és vannak káprázatos, ezt kijátszó megoldások is.

## A(z adat)bank

A tranzakciók kezelésére az eddig elmondottak alapján az 5. ábrán vázolt adatbank képzelhető el. Az adatbank lényegében a 3. ábrán látható modell számítógépes megfeleltetése. Mint látható, az egyes részfolyamatok adattömegeit különböző adatbázisok tartalmazzák. Az adatbázisok adatelemeinek kapcsolatát olyan adatbázis-kezelők gondozzák, melyek ezen relációkat vagy fizikai címeken, vagy hatékony indextechnikával írják le.

**Starcz Andor**




*5. ábra*

## Kiváló áruk fóruma — hogyan pályázhatunk?

# „Tesztkontroll"

Szoftvertéka rovatunkban egy olyan Microsoft-termékkel is foglalkozunk, mely az automatikus tesztelés hivatott bajnoka. Cikkünk szerzője megragadta az alkalmat, és egy tesztútmutatót is mellékelt az MTW ismertetésének ürügyén, mely azonban nemcsak azoknak a szakembereknek hív(hat)ja fel a figyelmét e fontos procedúra mozzanataira és eszközeire, akik a szoftvert meg akarják/tudják vásárolni, hanem mindenkiét, aki a minőségben érdekelt.

Mielőtt egy felhasználóknak szánt alkalmazást kiadnának, az hosszú és összetett fejlesztési stádiumokon jut keresztül, úgymint: prototípus-készítés, prealfa, alfa, prebéta, béta, prerelease, és végül a piacképes késztermék. Az alfa és béta fokozatú tesztelőket csupán az különbözteti meg, hogy míg az alfások az eredeti fejlesztő alkalmazásában állnak, a bétások már nem. Az előbb említett változatok ugyanannak a szoftvernek több verzióját testesítik meg. Ebből következik, hogy egy terméket több tucatszor vetnek alá különböző tesztelési eljárásoknak. A kézi teszt itt már túl sok időt és pénzt emésztene fel — különösen igaz ez az állítás közepes vagy nagyszabású alkalmazások kihozatalára.

Egy kézenfekvő példa: egy szövegszerkesztőt kell tesztelni, hogyan viszonyul az Application Windows területén ejtett egérmanipulációkhoz. Kézzel végigjátszani minden lehetséges szituációt: ez szinte lehetetlen feladat elé állítaná a teljességre törekvő fejlesztőket. Azonban egy szisztematikusan felépített, a teljes képernyőterületet bejáró ciklikus teszt gyors és tökéletes megoldást ad. (Ilyenkor domborodik ki az automatikus teszt előnye a hagyományos, intuitív és kézi eljárásokkal szemben.)

### „Tesztünk titkai"

A legfontosabb, hogy konkrét, kimunkált tesztelési tervvel fogjunk hozzá a munkához. A terv alapvető elemei:

— a tesztelendő alkalmazás;

— mire terjedjen ki a teszt;

— a teszteljárások során alkalmazott segédeszközök köre, azok szerepe;

— a különböző tesztelési platformok, azok pontos specifikációja;

— a program minőségtesztjét meghatározó módszerek kiválasztása;

— alapkritériumok, melyek meghatározzák a forgalmazandó termék minőségét és alkalmazhatóságát;

— a termék lehetséges alkalmazási területei;

— a tesztsorozatok szabatos leírása;

— a felhasználandó tesztadatok köre;

— a különböző elméleti és gyakorlati küszöbértékek és a kivételes (extrém) feltételek, amelyeknek közepette a programnak nem szabad megbotlania;

— az egyes tesztfázisok pontos időrendi ütemterve, forgatókönyve (mikorra — meddig);

— a fejlesztői, marketing- és tesztelő-csoport pontos szerepének meghatározása a tesztelési időszak előtt, alatt és után.

Ha a fenti pontok mindegyikére kielégítő választ ad a tesztelési tervünk, a csatát már félig megnyertük.

### Előkészületek

Az előkészítés több okból is a legnehezebb aspektusa egy tesztelési procedúrának:

— Gyakorta ez a legidőigényesebb része a tesztnek.

— Messzemenően óvatos tervezést, struktúrát, szervezetet és figyelmet követel meg.

— Az „essünk túl rajta, hadd menjen" hozzáállás általában megbosszulja magát.

Ki kell választanunk a legmegfelelőbb tesztelési segédeszközöket is (igaz ez az MTW kínálta többféle megoldásra is). Ügyelnünk kell a rendszerfüggetlen, körültekintő tesztre (például egy program, ami elszáll EMS memóriabővítés esetén, még kiválóan működhet extended memóriával):

— Életszerű helyzeteket teremtsünk, vegyük elő a legkülönbözőbb különc drivereket, tárrezidens programokat, diszk cache szoftvereket, hálózati meghajtókat, scannereket stb.

— Tüzetesen vizsgáljunk meg minden elképzelhető memóriakonfigurációt: EMS, extended- és csak konvencionális RAM jelenlétében is teszteljünk, változtassuk az elérhető memória méretét, például 2, 4 vagy 8 Mbájt extended memóriával is érdemes próbálkozni.

— Használjunk eltérő típusú matematikai társprocesszorokat és alaplapokat.

— Vizsgálódjunk CGA, EGA, VGA, Super VGA kártyás gépeken egyaránt.

— Használjunk minél többféle szabványt megvalósító nyomtatót.

— Különböző multitasking környezeteket se felejtsünk el létrehozni, próbáljunk a háttérben DOS programokat futtatni.

### Tesztfázisok

Figyelmesen dolgozzuk fel, analizáljuk a kapott tesztnaplókat és egyéb kimeneti állományokat. Ne felejtsünk el a tesztelés során bármilyen, a forráskódon belüli változtatást tisztességesen dokumentálni. (Például az MTW-hez ügyesen megírt tesztscript képes öndokumentálásra is!) Egyszerre csak keveset változtassunk, ne vonjunk össze több lépcsőt!

Ne feledkezzünk meg a teszt tesztjéről sem. Alkalmazzunk szisztematikus hibakeresést. A naplóállományokból kiszűrhető egy hiba gyakorisága; a periodikusan ismétlődő hibaminták (rekurzió gyanúja); a hibák összefüggése. Egyetlen hiba több másikra vezethet, amelyek normális működés esetén fel sem lépnének.

Ha bármilyen rendellenességet észlelünk, kinyomozandó annak oka. Hiba

a programban? Több extrém feltétel egybeesése? Az alkalmazott hardver vagy szoftver környezettel kapcsolatos inkompatibilitás? A program egy másik részének rendellenes működése?

Tanácsos egy ún. termékadatbázist vezetni, melynek karbantartása minden tesztelőre, fejlesztőre nézve kötelező. Érdemes feltüntetni:

— egy rövid címet (esetleg a hiba sorszámát);

— a hiba teljes leírását;

— a tesztelt program verziószámát;

— a tesztkörnyezet részletes leírását (hardver és szoftver egyaránt);

— a hiba komolyságát, minősítését (kritikus hiba, esztétikai hiba, programozási hiba, tervezési korrekció szükséges, az adott rész nehezen érthető);

— a hiba prioritását a kijavítás szempontjából;

— ki észlelte;

— ki javította ki;

— a kijavított programváltozat azonosító száma;

— a hiba mostani állapota;

— kinek a felelőssége/feladata a hiba kiküszöbölése.

## Kollektív tesztek

A fent leírt módszerek kisebb munkacsoportok esetén kiválóan működnek. Növeli a hatékonyságot, ha az emberek hálózatos gépeken dolgoznak, ugyanis:

— A teszt központilag vezérelve egyidejűleg több terminálon is futhat, amelyek egyenként lehetnek eltérő hardver- és szoftverkonfigurációjúak.

— Az egyes tesztelők és programfejlesztők közötti kommunikáció megoldható elektronikus postán keresztül.

— Béta-tesztelésnél előszeretettel alkalmaznak elektronikus postaládát — így a tesztelőnek lehetősége van megosztani észrevételeit a fejlesztőkkel és más béta-tesztelőkkel.

Reméljük, a fentiek előbb-utóbb itthon is elterjednek a gyakorlatban (talán épp az MTW tereli progresszívabb irányba a hazai tesztkultúrát), és a magyar fejlesztők egyre több kiváló, csontnélküli programmal örvendeztetik meg a felhasználók programra éhes, ezerszemű, állandóan elégedetlen és kritikus táborát.

**Herczeg József**

## Két dudás egy csárdában

Amikor a nyáron bejelentették az 1992. évi Compfair kiállítás időpontjának előrehozatalát, igen nagy felzúdulás fogadta a hírt. A szakma kiemelkedően legfontosabb hazai eseménye kényszerült kitérni egy ugyancsak rangos nemzetközi vándorkiállítás — a Telekom — elől, merthogy az utóbbi nem kívánta magát azonos időpontban megrendeztetni az előbbivel, ugyanakkor ragaszkodott október második hetéhez. Így hiába volt a Compfairt szervező Compexpo minden érve, az „erősebb kutya" elve alapján az ENSZ- és kormánytámogatást élvező Telekom keresztülvihette akaratát. Szó érte a mindkét rendezvénynek otthont adó Hungexpo házatáját is, mondván: a másik nagy itthoni számítástechnikai kiállítás, az Ifabo sikerében sokkal inkább érdekelt Hungexpo sem lépett fel kellőképpen a Compfair védelmében.

A meglehetősen borús előjelek ellenére — úgy tűnik — a Compfairt nem sikerült megrendíteni, továbbra is élvezi a szakma bizalmát. (Jól illusztrálja mindezt a Compfair kapcsán mostani lapunkban található számos hirdetés.) Mindannyian jobban örültünk volna az eredeti tervek szerinti közös Compfair-Telekom kiállításnak, hiszen a kétféle profil szervesen összetartozik. Úgy véljük azonban, hogy — feledve a rossz szájízt hagyó előzményeket — a két rendezvény külön-külön is jól szolgálhatja szakmai kultúránkat, az ott szövődő üzleti kapcsolatok pedig lendíthetnek valamicskét az egész ipar zötyklődő szekerén.

Minden érdeklődőt vár tehát október 6-tól 10-ig a Compfair, az azt követő héten, október 12-től 17-ig pedig a Telekom. S bár az egymást követő kiállítások építése-bontása miatt a BNV területén a szokásosnál nehezebben lehet majd megközelíteni a csarnokokat, reméljük, hogy a lényeg, a felvonultatott hardver- és szoftverkínálat mindenkit kárpótol ezért a kényelmetlenségért.

## E számunk hirdetői


| | Info# | Oldal |
|---|---|---|
| Allegro | 03 | 13. |
| ANT | 66 | 13. |
| Areco | 26 | 41. |
| Axis | 40 | 53. |
| Barex | 52 | 72. |
| Beco | 41 | 53. |
| Bürotech | 49 | 65. |
| Cédrus Rt | 50 | 66. |
| Cédrus Rt | 34 | 58. |
| Comex | 72 | B/III. |
| Comfort | 46 | 58. |
| CompMark | 37 | 52. |
| CompuDrug | 36 | 52. |
| ComputerBooks | 74 | 81. |
| Computerland | 73 | B/IV. |
| Computer Praxis | 62 | 76. |
| Controll | 58 | 75. |
| Corg | 27 | 43. |
| Corwell | 59 | 76. |
| C. Rex | 64 | 81. |
| Data Doctor | 53 | 72. |
| Datentechnik | 12 | 35. |
| Daxon | 11 | 35. |
| Déma | 54 | 72. |
| E-Copy | 67 | 13. |
| Elender | 56 | 73. |
| Elin | 02 | 01. |
| Euro-Profil | 63 | 12. |
| Fan | 09 | 35. |
| Floppyland | 28 | 43. |
| 3M | 01 | B/II. |
| Hoktrade | 42 | 55. |
| Holland Rt. | 20 | 37. |
| Holland Rt. | 21 | 40. |
| Holland Rt. | 22 | 42. |
| Holland Rt. | 23 | 42. |
| Hun-Comp | 35 | 52. |
| Identik | 17 | 36. |
| IQ Stúdió | 61 | 76. |
| IR Szerviz | 08 | 28. |
| ITEA | 47 | 58. |
| Kerszöv | 04 | 82. |
| Keszo | 06 | 26. |
| Kürt | 33 | K/IV. |
| Macroda | 48 | 65. |
| Mag ICS | 75 | 61. |
| Makrotrend | 45 | 79. |
| Megamicro | 60 | 76. |
| Miki | 57 | 74. |
| Minibit | 13 | 35. |
| ModiSoft | 65 | 81. |
| Netrend | 30 | 44. |
| Offi-Comp | 10 | 35. |
| Onyx | 31 | K/IV. |
| Pentacomp | 05 | 20. |
| Peron | 68 | 13. |
| Precomp | 43 | 55. |
| Qwerty | 55 | 73. |
| Rezon | 29 | 43. |
| Rolitron | 44 | 55. |
| SCI Modem | 24 | 39. |
| Server | 32 | K/IV. |
| Sol-Info | 25 | 41. |
| Spectral | 38 | 52. |
| Szilicium | 07 | 27. |
| Szint | 16 | 36. |
| Szoftver ABC | 71 | 89. |
| Telehold | 19 | 36. |
| Toner | 18 | 36. |
| Unitrade | 15 | 36. |
| Unitrade | 69 | 89. |
| VT-Soft | 51 | 72. |
| Wach | 14 | 35. |
| X-Byte | 70 | 89. |
| Xfer | 39 | 54. |

# Hírzárlat

Augusztusi számunkban rövid, de annál felelőtlenebb ígéret látott látott napvilágot a Novell Network Universityről szóló beszámolóra vonatkozóan. Sajnos az előadások teljes információs anyagát a szeptemberi lap lezárásáig nem sikerült megkapni a Novell cégtől. Így ez a beszámoló csak azokra az újdonságokra szorítkozhat, amelyekkel a cikk szerzőjének személyesen is alkalma volt megismerkedni.

Nézzük először a tényeket. A Novell GmbH szervezésében június 22. és július 10. között előadók hada „rohanta le" Európa 16 nagyvárosát, hogy az érdeklődők számára ismertessék a cég legújabb termékeit. A hat témakör, melyekből 2-2 előadást tartottak, a következő volt:

1. Nyílt rendszerek integrációja NetWare, TCP/IP és NFS segítségével
2. Lokális hálózatok analízise — hibafelderítés a hálózatban
3. X.25 szolgáltatások és a nyílt rendszerek integrációja
4. Hálózatmenedzsment
5. LAN-IBM integráció, aszinkron hálózatok, X.25, ISDN
6. Novell hálózatok és a Unix

Az előadásokon ismertetett termékek mindegyike vadonatúj volt, sőt néhány majd csak ősszel jelenik meg a piacon. Az előadók azon túl, hogy a szakterületet kiválóan ismerték, showmanként is megállták a helyüket. És bár az előadások felépítése inkább az amerikai menedzserréteg ízlésvilágát tükrözte vissza, az igényesebb és inkább a műszaki megoldásokra kíváncsi magyar hallgatóság kérdéseire a szünetekben lezajló beszélgetéseken az előadók készségesen és részletesen válaszoltak.

Sok esetben nagyon hiányzott az előadásokon elhangzottak gyakorlati alkalmazásának demonstrációja.

A legfrissebb adatok szerint szerte a világon 1 millió Novell szerveren és 15 millió munkaállomáson működnek a Novell 8. generációs programjai. Egyébként az év végére ígérik a NetWare 4.0-t, ami már a 9. generációt képviseli.

Ahogy az az előadásokból is kitűnt, a Novell erősíteni kívánja piaci pozícióját nemcsak a PC-s hálózatok területén, de — felismerve a kor szavát — a különböző rendszereket magába foglaló heterogén hálózatok piacán is.

Sajnos csak diákon mutatták be, de nagyon hatásos volt az új LAN WorkPlace for Windows, amely kiegészítő modulokkal lehetővé teszi egyetlen munkaállomásról Novell és Unix szervereken elhelyezkedő fájlok kezelését, valamint IBM nagygépekre való belépést 3270-es terminálemulációval. Mondjuk, egyetlen egérmozdulattal másolni fájlokat egy Sun és a Novell szerver között — nem kis dolog.

Ugyanilyen látványos volt az új hálózatmenedzsment program bemutatója is. A természetesen Windows alatt futó kezelői felületen keresztül a hálózat minden gépéről részletes információt lehetett kérni, részben a statikus adatbázis alapján, részben pedig a munkaállomásokon működő „menedzsment-ügyvivőkön" keresztül, amelyek a futó programoknak az aktuális memóriakiosztásáról, valamint a rendszer erőforrásairól nyújtanak információt.

Mivel rendszergazdaként magam is részt vettem egy hálózatmenedzselő program megálmodásában, kíváncsian várom az első gyakorlati tapasztalatokról szóló beszámolókat. Diaképeken keresztül mindenesetre lenyűgöző volt a program tudása.

**Villányi László**

UFF-híradó

# A szokott helyen és időben

A nyári szabadságok megkezdése előtt — a nagy meleg ellenére is — szép számmal jöttek érdeklődők a Unix fórum elektronikus levelezéssel foglalkozó összejövetelére. Eddig elég szűk körben (akadémikusok, kutatók) használták ezt a szolgáltatást. Ma már azonban többféle levelező rendszerhez (Ella, Petra, Elv) is hozzáférhetünk, feltéve, ha előfizetünk rá. A levelezés ugyanúgy történik számítógéppel, mint hagyományos módon. Küldhetünk egyetlen személynek (gépnek) levelet, de körlevél írására is van lehetőségünk. Mivel a levelek a fa struktúra elve szerint „közlekednek" a hálózatban, a levél optimális útvonalon jut el a címzetthez.

Magyarországon még mindig elsősorban a kutatók választják a levelezés elektronikus formáját. Távoli baráttal, kollégával ilyen módon írják közös cikkeiket. Ez az új szolgáltatás lehetővé teszi, hogy a levelező rendszereken keresztül megkapják az őket érdeklő programokat, dokumentációkat, szabványokat, mérési adatokat.

Tudnunk kell azonban, hogy bizalmas információk átvitelére nem megfelelőek a levelező rendszerek, hiszen kis ügyességgel kívülállók is elolvashatják a leveleket. Nincs semmilyen garancia arra, hogy titkosak maradnak (vagy nem vesznek el) az információk. Viszont minden olyan helyen, ahol közlendőink nem bizalmasak, továbbá nem számít, hogy levelünk egy-két órát vagy esetleg egy napot is késik, jól használhatjuk ezt az újfajta szolgáltatást.

Magyarországon az IIF program keretében már 1986-ban létrehoztak egy olyan hálózatot, amelyen hazai kutatók tudtak információt cserélni egymással, továbbá különféle adatbázisokhoz is hozzáférhettek. Ezek közül ma is több adatbázis-szolgáltatás széles körben és ingyenesen használható.

Így a szöveges adatbázisok 99%-ába térítés nélkül bepillanthatunk. Ilyen például a szótörténeti tár, a disszertációk és a kutatás-fejlesztés adatbázisa, a szabványügyi adattár, a ki-kicsoda, az egészségügyi és a humán tudományok adatbázisa. Égető szükség lenne azonban egy céginformációs és egy hatályos jogszabályok gyűjteményét tartalmazó adatbázisra is, hogy az információk ezen a területen is naprakészek legyenek.

Az UFF nemcsak ezekkel a sokunk számára sajnos még elérhetetlennek tűnő lehetőségekkel ismertetett meg bennünket, hanem az egyik felhasználó beszámolt egy konkrét alkalmazás tapasztalatairól is. S hogy ne szakadjunk el teljesen a levelek világától, a Postánál alkalmazott, teljesen automatizált raktár-, készletgazdálkodási és pénzkezelési rendszer (Protheus) lehetőségeit ismerhettük meg. Az előadó nem titkolta elégedettségét, csak dicsérni tudta a kialakított rendszert. Szerinte a szoftver minden olyan támogatást megad, amely napi munkájukhoz szükséges. A felhasználóbarát program mindennapos alkalmazásával sokkal kevesebb munkájuk lett az ott dolgozóknak, mint addig volt. S mi más is lehetne egy applikáció feladata, mint hogy segítségével gyorsabban és pontosabban végezhessük dolgunkat.

A Unix Felhasználók Fórumán továbbra is érdekes témák kerülnek terítékre. Így az őszi klubindító összejövetelen beszámolót hallhatunk a szeptember 28—30-a között megrendezendő HUUG konferenciáról, valamint a Compfair tapasztalatairól. S hogy a hagyományokhoz hűek maradjunk, nem hiányozhat egy konkrét Unix-alkalmazás bemutatása sem.

**Szeibig Andrea**

Az UFF első őszi randevúja október 14-én lesz a szokott helyen és időben:
Cédrus Rt., Bp. XI., Karolina út 17., Konferenciaterem, 15 óra.



INFORMÁCIÓKÉRÉS: 24 ▶
INFORMÁCIÓKÉRÉS: 42 ▶
INFORMÁCIÓKÉRÉS: 44 ▶

## Házi földrengésjelzők?

Amerikai földrengéskutatók és számítástechnikai szakemberek azt javasolják, hogy a San Francisco-i öböl térségében önként vállalkozók otthoni személyi számítógépeihez 500-1000 dollárba kerülő kis földrengésjelző műszert kapcsoljanak. Az ezekben feljegyzendő adatokat az Egyesült Államok Geológiai Felügyeletének kaliforniai központjába továbbítanák.

A kutatók becslése szerint e központ 100 kilométeres körzetében százezer család kapcsolódhatna be a kísérletbe. Még ha erre csak egyetlen százalékuk vállalkozik is, és az adatoknak csak a fele bizonyulna megbízhatónak, akkor is a rengésjelzők minden eddiginél sűrűbb hálózatát alakíthatnák így ki. Ez a hálózat a múltbelieknél tízszerte gyengébb rezgéseket is jelezné, megkönnyítené megjelölni azokat a helyeket, ahol különösen nagy a földrengésveszély, s egyedülállóan gazdag adathalmazt szolgáltatna arról, hogy miként terjed egy földmélyi törés hatása.

## A történelmi magánélet elemzése

A kasseli egységes főiskolán Heide Wunder professzor és csoportja érdekes vállalkozásba kezdett: 1380 és 1730 között élt lipcsei házasemberek életútját dolgozzák fel, s ehhez igen eredeti módszert találtak ki. Halotti beszédekből gyűjtött információkat értékeltek ki egy erre a célra írt grafikai programban. A halotti prédikációkban szóba kerülő sokféle kapcsolatot vizuálisan ábrázolják: például a rokoni szálakat, a helyváltoztatást, az életutat és érvényesülést. Eddig 102 prédikációt vittek fel a számítógépre, és ebből 2200 emberre vonatkozó információkat találtak. A történészek célja az, hogy jól megalapozott képet kapjanak a „polgári házasság" intézményének létrejöttéről.

## Golyó lyuggatta számítógép

A bostoni számítógép-múzeum volt az első, amelyet kizárólag az elektronikus gépeknek és a társadalomban elfoglalt helyüknek szenteltek. A kiállított darabok többsége alig öt-tíz éves, de mivel a számítástechnika fejlődési üteme szédületes, ezek teljesítménye a maiakéhoz képest valóban eltörpül.

A múzeum nemcsak művelődéstörténeti missziót vállal, hanem az is a célja, hogy megkedveltesse az emberekkel e gépeket. Oliver Strimpelnek, a múzeum restaurátorának Walt Disney-hez méltó ötlete támadt: egyévi munkával olyan hatalmas számítógépet készített, amelynek belsejében sétálni lehet. Akkora, mint egy kétemeletes ház, benne az embernek olyan érzése támadhat, mint az összezsugorodott Nils Holgersonnak. Strimpel szerint a monstrumnak demisztifikáló hatása van.

A múzeum történeti része sem akármilyen. Ezerötszáz gépet állítottak ki, köztük azt a Hollerithet, amelyet az 1890-es amerikai népszámláláskor alkalmaztak. Itt van a NEC, amely az Apolló-missziót vezette a Holdra. A látogatók megismerhetik az Univac-1-et, az első amerikai kereskedelmi számítógépet, amely 1951-ben megjósolta Eisenhower következő évi elnökválasztási győzelmét. Egy sorban áll a Cray-1, az első szuperszámítógép a legnagyobbal, a dinoszaurusznak tűnő Whirlwinddel, amelyet 1945-ben készített az MIT az amerikai haditengerészet számára. A múzeum egyik kincse a WISC, amelyet egy lőtér mellett használtak az ötvenes években — ennek testét golyók lyuggatták.

Milyen gép kerülhet be a múzeumba? Amelyik a maga műfajában elsőként vagy utolsóként, klasszikussá válhat, és üzemképes állapotban van.

## Oscar-díjas mikrogépek

A számítógépek „Rolls-Royce"-a hazánkban is sikert arat, bizakodik a forgalmazó, a Westimp Kft. Mint a Goupil — az egyik legnagyobb francia elektrotechnikai és telekommunikációs konszern — hivatalos magyarországi értékesítője a teljes kínálatot a hazai érdeklődők rendelkezésére kívánja bocsátani. És milyen áron? Nos ez azonos a németországiéval — ígérik a forgalmazók. A formatervezés Oscar-díját is elnyert gépek elsősorban az igényesebbek érdeklődésére számíthatnak.

## Számítógépes zokni

A Zalareform Kft. újpesti gyárában számítógépes zoknikat készítenek. Mit jelent ez? A számítógépes zokni jellegzetessége, hogy mintája korlátlanul variálható. Ezt az európai színvonalat képviselő 20 db Irmac 520 típusú zoknigyártó gép teszi lehetővé. A 60 százalékban pamut alapanyagú zoknik számítógép-vezérlésű mintaváltással és kötéssel készülnek. A gépek teljesítménye 780 ezer pár évente, amit további hat géppel egymillió pár fölé emelnek.

## Suzy

Ez a programrendszer két részből áll: a PerSpectoGraph háromdimenziós, interaktív, mozgóképes grafikus rendszerből és az akár Turbo Pascal 6.0-ból is hívható Graph3D háromdimenziós grafikus unitból.

A programrendszer tetszőleges térbeli testet tud ábrázolni. A megjelenítés módja perspektív. Tulajdonképpen az emberi szem szimulációs modelljét valósítja meg, azaz az emberi szem recehártyáján megjelenő kép látható a számítógép képernyőjén is.

A térábrázolást tulajdonképpen három fő paraméter határozza meg, ezek a szem, az általa nézett pont és a fényforrás háromdimenziós koordinátái. Ezekre épül a program is, mégpedig oly módon, hogy e paraméterek rugalmasan módosíthatók legyenek. Ez igaz más, szintén fontos paraméterekre is, például a takarásra. Ki- és bekapcsolhatjuk a takarási mechanizmust. Az egyik állapotban a szemlélőhöz képest a test takart részei is láthatóak, a másikban nem. A test tehát átlátszóvá tehető.

A leendő rajzot nagyíthatjuk, kicsinyíthetjük, eltolhatjuk, ezáltal a képsíkon egy tetszőleges ablakot vághatunk. Az ábrázolt test foroghat és bármely irányban mozoghat a képernyőn egy saját fejlesztésű számítógépes videotechnika segítségével. A film végtelenített, végigjátszása után újrakezdi, ezért nagyon alkalmas egy alakzat forgásának bemutatására.

A Turbo Pascal-könyveiről ismert szerző, Pirkó József által kifejlesztett Suzy érdekessége, hogy a programok egy részét a szintén a szerző által létrehozott Judy nevű fejlesztőrendszer írta meg.

## A műhold és a vegyipari szakszervezet

A Vegyipari Dolgozók Szakszervezeti Szövetsége (VDSZ) is használhatja a Vegyi, Energia és Általános Ipari Dolgozók Szakszervezeteinek Nemzetközi Szövetsége (ICEF) műholdas, számítógépes hálózatát.

E rendszeren — napi 24 órán át — nemcsak közlések áramolhatnak, de az ICEF tagszervezetei (köztük már a VDSZ-szel), hozzáférhetnek az ICEF több mint kétezer adatbázisához. Azonnali információcserét biztosít a rendszer, de emellett globális szolidaritási hálózatot is alkot. Használata sokkal olcsóbb, mint a kommunikáció megszokott formái.

A VDSZ-t 1990-ben vették fel az ICEF rendes tagjai közé, ekkor ajánlották fel a távközlési hálózatban való részvétel lehetőségét. A nagy kérdés azonban, hogy — a technikai kapcsolat létrejötte után — hogyan tudják az itthoni szervezetek hasznosítani a rendszer előnyeit, miként élhetnek vele?

## Nemzetközi BBS Szlovéniában

Bár a volt Jugoszláviát még véres harcok dúlják fel, Szlovéniában már a műszaki fejlesztéssel foglalkoznak. Mariborban e nyáron hozták létre az Espnet nevű BBS-t. Hogyan működik? A rendszer csomópontokon alapul, amelyek bizonyos időközönként információt cserélnek egymással. E célra — modem közbeiktatásával — a telefonhálózatot használják. Abban az esetben, ha a csomópont a felhasználóhoz elég közel, sőt ideális esetben az adott városban van, a kommunikációs költség minimális, hisz legjobb esetben helyi telefonbeszélgetésnek minősül. Ekkor a felhasználó maga előtt a helyi csomópontot látja, amelybe a teljes Espnet hálózatból érkeznek a küldemények. A csomópontról csomópontra történő információ-átadás a felhasználó számára láthatatlan. Az Espnet kezeléséhez a felhasználónak csupán eszperantóul kell tudnia. (Az Espnetről részletesen tájékoztat az Internet „BHERMAN@UNI-MB.AC.MAIL.YU" és a Fidonet „2:380/104".)

## Nyelvi morfondírozások

# Ne kenjük a gépre!

Mármint azt ne kenjük a gépre, ha számítógéppel írott, szedett és kinyomtatott szövegeink nyelvtanilag és helyesírási szempontból hibásak. Vannak ugyan a számítógép használatának sajátosságaiból eredő jellegzetes hibaforrások, de azok mögött a hibák valódi oka rendszerint saját fogyatékos nyelvtudásunk vagy a korrigálás elmulasztása, s csak az esetek kisebb hányadában az, hogy beleütközünk az adott pillanatban áthidalhatatlannak bizonyuló hardver- vagy szoftverkorlátba.

### Sajtóhiba

Ez a legegyszerűbb, de egyben a leginkább fatális jelenség, mert minden elképzelhető kontroll ellenére bekövetkezik. Szerencsére az olvasó is ösztönösen ráérez, hogy ilyen esetekben nem az írástudók hiányos műveltségéről vagy lelkiismeretlen hanyagságáról van szó, hanem valamilyen szokatlan „malőrről", ami azonban ennek ellenére nagyon kellemetlen lehet.

Az Alaplap 2 év alatt nem produkált egyetlen igazán feltűnő vagy kínos sajtóhibát sem, ezért Murphy törvényei szellemében már esedékes volt, hogy történjen ilyesmi is. Hát tessék, itt van, nézzék meg e számunk lemezmellékletének feliratát: azon a „PC-RE" helyett „PC-RL" olvasható. „Nem vészes!" — nyugtatgattak minket. De azért bosszantó. A története viszont érdekes.

A floppymelléklet fizikai alapanyagát szállító cégekkel tárgyalva az eddigi legeredetibb ajánlatot a KAO cégtől kaptuk: nem kell külön címkét nyomatni és azt felragasztani, hanem minden információt közvetlenül rányomtatnak a lemez védőburkolatára. A magyarországi márkaképviseletet ellátó Makrotrend elkészíttette a számítógépes grafikát, azt kiküldték Kanadába, ott pedig a KAO-nál szitanyomással rányomtatták mind a 11 000 szeptemberi lemezre. A CorelDraw-val készült grafikus fájl itthon még teljesen hibátlan volt, a gyártás során Kanadában keletkezett egyetlen betűnyi eltérés okát pedig lapzártáig nem sikerült kiderítenünk. (Mindenesetre rejtélyes, hogy ha konverziós vagy más okból bele kellett javítani a fájlba, az ékezetes betűk hogyan maradtak hibátlanok és miért az egyszerű E(nglish)-t törölték Le.)

### Gépnyelv

Itt most nem a gépi kódra gondolunk, hanem arra a sajátos „hunglish" vagy „mangol" vagy „angyar" nyelvre, amely az angol nyelvnek a programkészítésben kivívott egyeduralmából és anyanyelvünk ösztönös használatából ötvöződik, a mindenkori hardveres és szoftveres korlátokhoz igazodva. Bármily meglepő, ma is készülnek a nagygépes hőskorszakra jellemző, teljesen ékezet nélküli faramuci „magyar" szövegek. Néha még cikkeket is kapunk ilyen 7-bites ASCII formátumban. Persze nem ez a jellemző, hanem a 8-bites ASCII használata, csak kissé hevenyészve, slamposan, különösen ami a hosszú magánhangzók írását illeti.

Végiggondolva a sokféle szempontot — melyekről bővebben augusztusi számunkban, a hónap témája keretében írtunk —, jelen pillanatban egyetlen kompromisszum látszik életképesnek. Egyik oldalról nyelvi kultúránk védelmezőinek be kell látniuk, hogy nem lehet megkövetelni a korrekt magyar írásmód használatát ott, ahol az a számítógépes környezetben zavarokat okozna, kezdve az alkönyvtárak elnevezésétől a „szabadon" definiálható parancsokon át egészen a forrásprogramoknak a fordítóprogramok által értelmezendő szövegéig. Ezzel szemben a számítástechnikusoknak be kell látniuk, hogy semmilyen mentségük nincs arra, ha nem a komplett magyar ékezetes betűkészletet használják a közlésre szánt szövegekben, a forrásprogramok kommentáló részében, a működő programok üzeneteiben, a saját készítésű menükben, az adatbázisokban stb. Szívesen vesszük, ha valaki vitába száll velünk, és elfogadható érvekkel tudja alátámasztani, hogy esetleg máshol kell meghúznunk a „demarkációs vonalat".

### Kódtábla

A szövegek helyes magyarsággal való megjelenítése sok technikai feltételtől függ. Visszafelé kezdve, nyilvánvalóan elengedhetetlen kellék a nyomtató alkalmassága (hardver+szoftver). Az is fontos, hogy a monitoron lássuk is, amit beírtunk, a billentyűzet pedig „engedelmeskedjen" akaratunknak. A legfontosabb azonban maga a kód.

Ez így talán nem elég meggyőző, de a konkrét példa azzá teszi. Szerkesztőségünk szerencsére sok külső szerzőtől kap anyagot. Ahányan, annyiféle szövegszerkesztőt használnak, többnyire CWI vagy Ventura kódkiosztással. Nekünk mindegyik jó, még az is, amelyik teljesen egyedi, de van mellette egy kis „help", hogy az egyes ékezetes betűk mely ASCII kódokon lapulnak. Abból már rutinmunka megírni az általunk használt Kedit kódkonverziós makróját, és előállítani a Ventura számára emészthető szöveget.

Átkódolni azonban csak azt lehet, ami be van kódolva! Nem ritka eset ugyanis, hogy egy-egy terjedelmes cikkben egyetlen hosszú ő és hosszú ű sincs, helyükön ö és ü szerepel, gyakran hibádzik az ú, és a szerző a hosszú í-kkel is nagyon takarékosan bánik... Mit lehet ezzel tenni? Szerkesztés közben javítani, az nagyon deprimáló érzés; betűnkénti rákérdezéssel végigcserélni már vidámabb dolog, de nagyon favágó munka; a helyesírás-ellenőrzők pedig még nem elég termelékenyek.

Mi csak egy jó megoldást ismerünk: a szöveget rögtön úgy megírni, hogy minden betű helyesen szerepeljen benne, akármilyen módon, de a többitől megkülönböztethető kóddal. Azzal utána már mindent lehet kezdeni: lemezre tenni, mátrixprinteren kinyomtatni, kiadványszerkesztőbe betölteni... Egy a lényeg: ha nem korrekt a szöveg, azért ma már nem lehet a technikai feltételeket okolni. Magyar billentyűzetkezelő és képernyőbetű-letöltő bárki számára hozzáférhető, az ősi hardverekhez pedig van egyszerű módosítási lehetőség (pl. Eprom-csere).

Helyesírási hibáinkat nem szívesen ismerjük be. De azért ne kenjük a számítógépre!

**Faklen Pál**

INFORMÁCIÓKÉRÉS: 47 ▶

## Wolfram-szál

# A „klasszikus" Mathematica

A matematikai számítási feladatok megoldásának természetes eszközeként a számítógépet már a második világháború végétől alkalmazzák, hiszen első feladataként éppen katonai számításokra használták fel. A számítógépek „személyivé" válásával, és számítási teljesítményük gigantikus növekedésével ma már rendelkezésre állnak könnyen kezelhető matematikai programok. Ezeket mérnökök, tudósok, programozók és matematikusok használhatják számítási munkájuk megkönnyítésére.

A matematikai programok közül átgondolt tervezési koncepcióival, kitűnő teljesítményével (de árával is...) messze kiemelkedik a Wolfram Research cég által forgalmazott program, a Mathematica 2.0. Egy érdekesség: a Mathematica keresztapja az a Steve Jobs, aki az Apple, majd a Next számítógépeket megalkotta. Ő adta e nevet a szoftvernek.

A Mathematicát elsődlegesen az elméleti fizikában jelentkező matematikai problémák megoldására fejlesztették ki. Magas szintű programozási nyelvet tartalmaz, de interaktív módon is használható. A programról csak elragadtatással lehet szólni; nézzük meg, hogy miért. Számos számítógéptípuson képes futni: PC DOS- és Windows-környezetben, Macintosh gépeken és Unix-alapú munkaállomásokon (Sun, Next, ...) egyaránt. Ezt a sokarcúságot a program speciális és követendő, korszerű tervezési koncepciója biztosítja. Maga a program két részből áll: a számításokat ténylegesen elvégző magból, a kernelből, valamint a felhasználóval kapcsolatot tartó interfészből, az ún. „front end"-ből. A kernel alapjaiban minden fent említett számítógéptípuson azonos (mivel C-ben van megírva).

### Postscript mint segédeszköz

Egy másik jó döntés a tervezés és megvalósítás során az volt, hogy a grafikus adatformátumnak a Postscript nyelvet választották. A kernel és a front end a legegyszerűbb esetben egy karakteres, sororientált párbeszédes adatcsatornán kommunikál egymással: induláskor az In[1]:= bejelentkezéssel várja a kernel a végrehajtandó utasításokat, és az eredményeket is így jeleníti meg.

Például:

```
In[1]:=N[ Log[ 4 Pi ] ]
Out[1]:=2.53102
```

Nagyon lényeges, hogy a rendszerben használt függvények, parancsok és változók nagy kezdőbetűsek legyenek, mert különben a rendszer nem ismeri fel ezeket.

A Mathematica igen sok módon és sokféle célra használható:

1. Numerikus számításokra, szimbolikus alakokkal való számolásra és adatok grafikus megjelenítésére képes. Ilyenkor úgy működik, mint egy igen magas színvonalú kalkulátor. Számok esetén a számítási pontosság tetszőleges lehet (1. ábra).

A szimbolikus alakokkal történő számolás az algebra, az integrálás és differenciálás használatát biztosítja. Nagyon érdekes, hogy tud szimbolikus kifejezéseket egyszerűsíteni, hatványozni, egyenleteket összevonni, és egyenletrendszereket megoldani (2. ábra).

A mátrixműveletek is a repertoárjába tartoznak (3. ábra). Itt a „%" jel az előző kimeneti eredményt jelöli.

Grafikai képességei lehetővé teszik két- és háromdimenziós grafikák megjelenítését, függvényekét, adathalmazokét egyaránt, színesben is. A háromdimenziós ábrázolásoknál a valósághűséget az árnyékolás, színezés és a megvilágítási hatások biztosítják.

2. A C-hez hasonló programnyelven írhatók programok. Ezt a Mathematica interpreter módon kezeli: a program begépelés után azonnal futtatható.

3. Igen sok, matematikai táblázatokban lévő adat könnyen előhívható. Ismer számos átalakítási szabályt.

Például:

$(a+b)^2=a^2+2ab+b^2$;
log(xy)=log(x)+log(y); stb.

4. A Mathematica teljes munkakörnyezetet biztosít matematikai számításokhoz, programok futtatásához. A kernel minden számítógépen hasonló módon működik, a front end, a felhasználói

```
C>math
Mathematica (MS-DOS 386/7) 1.2 (June 7, 1990) [With pre-loaded data]
by S. Wolfram, D. Grayson, R. Maeder, H. Cejtin,
   S. Omohundro, D. Ballman and J. Keiper
with I. Rivin, D. Withoff and T. Sherlock
Copyright 1988,1989 Wolfram Research Inc.

In[1]:= 5+7

Out[1]= 12

In[2]:= 3^100

Out[2]= 515377520732011331036461129765621272702107522001

In[3]:= (3+4I)^10

Out[3]= -9653287 + 1476984 I

In[4]:= BesselJ[0,10.5]

Out[4]= -0.236648

In[5]:=
```

*1. ábra*

interfész határozza meg, hogy milyen lesz a program megjelenése. Grafikus felhasználói felülettel rendelkező számítógépeknél a Mathematica a „jegyzetfüzet megjelenítéssel” módszert alkalmazva szemléltet. Ennek az a lényege, hogy a számításokat interaktív módon végezhetjük el, és a jegyzetfüzet lapjain keverni lehet képleteket, grafikákat, hangokat és animációt. Ha egyre több a téma, lehet úgy is dolgozni a programmal, hogy a jegyzetfüzetben elolvashatók és megtanulhatók a témáról a tudnivalók, majd utána kezdhetjük a számításokat. Mivel a beépített szövegszerkesztő igen jó minőségű, a dokumentumokba kívánkozó megjegyzések szépen és gyorsan a helyükre kerülhetnek. A Postscript grafikus átviteli formátum biztosítja az összeköttetést a különféle típusú számítógépek között.

```
In[5]:= Permutations[{a,b,c}]

Out[5]= {{a, b, c}, {a, c, b}, {b, a, c}, {b, c, a}, {c, a, b}, {c, b, a}}

In[6]:= {x, y, z} . {a, b, c}

Out[6]= a x + b y + c z

In[7]:= Solve[ {a x + b y == 0, x - y == c} , {x,y} ]

                   b c           a c
Out[7]= {{x -> -(------), y -> ------}}
                 -a - b         -a - b

In[8]:= Integrate[a/(1-x^3),x]

                        1 + 2 x
         Sqrt[3] a ArcTan[-------]                                  2
                          Sqrt[3]      a Log[1 - x]    a Log[1 + x + x ]
Out[8]= ------------------------- - ------------ + -----------------
                    3                     3                6

In[9]:=
In[9]:=
```

2. ábra

## A többihez viszonyítva

Számos matematikai program a képletekben, kifejezésekben a szokásos nemzetközi matematikai jelöléseket használja, mint például a négyzetgyök és az integrál jele. Ez megnehezíti a programok és a számítási anyagok átvitelét más számítógépekre, mert azok interpretálása számítógéptípusonként változik.

A Mathematica ehelyett angol nyelvű, csak parancsokat tartalmazó szöveges kifejezéskönyvtárat használ a matematikai műveletek leírásához. Az egyenletek is ASCII formátumú szövegfájlokban vannak tárolva, ezért szintén könnyen átvihetők más rendszerkörnyezetbe.A Mathematica saját programnyelve több programnyelv szintéziséből épült fel. A Basichez hasonló az interaktivitás. A Fortranból átvett számos alapvető aritmetikai műveletet és kimeneti adatait is képes Fortran formátumban szolgáltatni. A Mathematica eljárás programozási módszere a C-ben lévőhöz hasonló, de a Pascal néhány eljárását is átvette. Ez mondható az APL programnyelvről is. A Mathematica nyelvezete legjobban az interpreter alapú Lisp programnyelvhez hasonlít. A grafikus adatformátum leíró nyelve a Postscript. A 2.0 verzió több mint 840 függvényt ismer, amely minden szokásos matematikai művelet elvégzését lehetővé teszi. Emellett adatbázis-kezelő és listázó funkciókkal is rendelkezik, valamint összetett függvényábrázolási és statisztikai analízist biztosító funkciói is vannak.

```
In[9]:= m = Table[ 1 / (i + j + 1), {i,3}, {j,3} ]

          1  1  1    1  1  1    1  1  1
Out[9]= {{-, -, -}, {-, -, -}, {-, -, -}}
          3  4  5    4  5  6    5  6  7

In[10]:= Inverse[ m ]

Out[10]= {{300, -900, 630}, {-900, 2880, -2100}, {630, -2100, 1575}}

In[11]:= % . m

Out[11]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

In[12]:= m - x IdentityMatrix[3]

           1      1  1    1  1       1    1  1  1
Out[12]= {{- - x, -, -}, {-, - - x, -}, {-, -, - - x}}
           3      4  5    4  5       6    5  6  7

In[13]:= Eigenvalues[ N[m] ]

Out[13]= {0.657051, 0.0189263, 0.000212737}

In[14]:=
```

3. ábra

A nyelv akár interpreter (kérdés — válasz), akár programfuttató üzemmódban képes működni. Az átvihetőség és a nagyon hatékony programnyelv mellett a különböző speciális tudományos részterületekhez kapcsolódó programok is használhatók kiegészítő programcsomagok formájában. Ezek az adaptációk Mathematica-programok, amelyeket specifikus feladatok megoldására írtak. Ilyenek állnak rendelkezésre a lineáris algebra, a számelmélet, a statisztika, a geometria területén.

Az ilyen programok folyamatos bővülését a Mathematicát alkalmazó munkacsoportok munkája biztosítja. A Windows alá készített változat grafikus megjelenése nagyon hasonló a Macintosh és a Next gépeken futó verziókhoz. A Mathematica kiterjesztett üzemmódban fut, felhasználva a virtuális memória nyújtotta előnyöket. Minél nagyobb a rendszer memóriája, annál hatékonyabban működik a program.

A programról mindez csak egy kis ízelítő. Használatának elsajátítása a matematikai ismeretek mellett sok kitartást és számítógép melletti tanulást igényel. Angol nyelven van irodalom is, melyet ajánlhatunk: Stephen Wolfram: Mathematica — A System for Doing Mathematics by Computer (Addison-Wesley Publishing Company, Inc. 1988. ISBN 0-201-19334-5).

A program DOS és Windows alatt futó változata is beszerezhető Magyarországon. Birtokbavétele javasolható mindazoknak a kutatóknak, mérnököknek, akiknek munkájuk során matematikai problémákat kell megoldaniuk. Egyetemistáknak sem jön rosszul... Nem fognak csalódni!

**Kónya László**

# Mag ICS

Informatikai Rendszerfejlesztő és Marketing Kft.

Compfair — Mag ICS-stand:
A pavilon
201/D




ICS Identcode-Systeme

Ajánlataink

**Vonalkódnyomtatók** — több mint 10-féle változat, mobil, nagy teljesítményű ipari, A4-es méretű, olcsó... A Mag ICS szakemberei által kifejlesztett szoftverek a legbonyolultabb címkék tervezését is lehetővé teszik.

**Vonalkódolvasók** — az „egyszerű" fényceruzától a nagy teljesítményű ipari lézerscannerig, mágneskártya-olvasó, résolvasó, pultscanner.

**Kézi adatgyűjtők** — leltározáshoz, mobil adatgyűjtéshez, a legkülönbözőbb időjárási körülmények között is működő berendezéseinkhez szakembereink által fejlesztett programokat is ajánlunk.

**Pénztárgépes kiskereskedelmi SW programcsomag** – pénztárosi funkciók, szállító-vevő nyilvántartás, leltár, ajánlatok gépi feldolgozása, rangsorolása, számítógépi levelezés-számlázás, percrekész információk gazdasági döntések meghozatalához, közvetlen kapcsolat az elosztó raktárakkal.

**Mérlegek** a vonalkódos rendszerben.

**Áruvédelem** — áruházi lopások megelőzésére szolgáló kapuk és védelmi jelzőcímkék (üzleteknek, könyvtáraknak).

**Kamerás védőrendszer** — bankok számára ajánljuk a behatolást, rendkívüli eseményt érzékelő riasztórendszerünket.

**Regisztrációs és beléptető rendszer** – bankok, nagyvállalatok, rt.-k nagylétszámú részvényes üléseinek vonalkódos regisztrációs és beléptető rendszerét szakembereink fejlesztették ki. A rendszerben alkalmazott olcsó laminált kártya törzsvevőkártyának, kölcsönzőjegynek, gyári belépőnek, hitelkártyának egyaránt alkalmas.

**Mag ICS PC** — tesztelt, jó minőségű 386-osok olcsón.

**Termeléskövetés, minőségbiztosítási rendszerek** — a Mag ICS szakemberei olyan integrált rendszerek tervét is ajánlják, amellyel vonalkódos, automatizált termeléskövetés és minőségbiztosítás valósítható meg. Referenciafilmünket ajánljuk a részletekhez.

**Címkék, festékszalagok** — több mint 2000-féle anyagot ajánlunk vonalkódos rendszereinkhez. Mivel a Mag ICS az ICS családdal együtt (ez 12 európai vállalatot jelent) adja le éves megrendelését, áraink 50%-kal alacsonyabbak a hazai árszínvonalnál.

**Minden Magyarországon eladott vonalkódnyomtatóhoz szállítunk címkét és festékszalagot.**

**Aki éves megrendelését egyben, előre jelzi, 50%-os árengedményt kap!**


**Mag ICS Kft,**
H-9400 Sopron, Bástya u. 75. • Tel.: (36)-99-14-250, (36)-99-34-035 • Fax: (36)-99-14-250


INFORMÁCIÓKÉRÉS: 75

## PADSor II.

# Kényelmesen, profi módon

Az első „PADSorban" megismertük a PADS felépítését, az alkatrészkönyvtár szerkezetét, valamint a PADS-Logic és a PADS-PCB modulok működését. A hátralevő „PADSorban" áttekintjük az automatizált tervezést biztosító elhelyező és huzalozó modulokat, a tervezőrendszer használatát még kényelmesebbé tevő különféle kiegészítő programokat, valamint a professzionális igényeket is kielégítő PADS-2000 rendszert.

A PADS PCB moduljába integrálható a PADS-Place automatikus alkatrész-elhelyező program, amellyel üres, illetve az alkatrészek egy részét tartalmazó kártyák teljes pozicionálása végezhető el.

### Elhelyezve

Az alkatrészek elhelyezhetők a PCB modulban definiált pozicionálási mátrixon (analóg kártyák esetében mátrix nélkül). Egy kész elrendezés azonban tovább finomítható a szomszédos elemek iteratív cseréjével.

Az alkatrészek könyvtárbeli leírásából tudjuk meg, hogy az adott alkatrész tartalmaz-e azonos, egymással felcserélhető funkciókat, kapukat vagy kivezetéseket. Amennyiben az elhelyező program lehetőséget talál az automatikus kapu- és kivezetéscserére — és a felcseréléssel kedvezőbb helyzetbe kerülünk —, akkor elvégzi azt. E cserék természetesen manuálisan is megvalósíthatók.

A végleges elhelyezés kialakulásával célszerű az alkatrészeket a kártyán elfoglalt helyük alapján átsorszámozni. Így a kártya már „rendezett" lesz, az alkatrészeket könnyű megtalálni. A cserék utáni állapot — az ECO fájl segítségével — a kapcsolási rajzon is megjelenik.

### Huzalozva

Hogy a mindenkori elhelyezés eredményes volt-e vagy sem, arról tájékozódhatunk a becsült összeköttetési hossz értékéből, a vezetékezés-sűrűség hisztogramból, valamint a színes sűrűségtérképből.

Az alkatrészek elhelyezése után következik a huzalozási fázis. A PADS-PCB modul alkalmas az egyes huzalok kézi bekötésére, azonban ezt az igen fáradságos munkát automatikus huzalozó programokkal is elvégeztethetjük. Ezt a legegyszerűbben a PADS-PCB menürendszerébe illeszkedő PADS-Route modullal tehetjük meg.

A Route modul úgynevezett Costed Maze algoritmussal dolgozó általános huzalozó program, amely 17 huzalozási fázist hajt végre. A fázisok a megengedhető legmagasabb „szabálytalansági" fokot határozzák meg. A legegyszerűbb fázisokban kizárólag rövid, egy irányban haladó jelek köthetők be, míg a legbonyolultabb tetszőleges számú rétegközi átvezetést (via) és irányítást tartalmazhat. A megengedett keretek között előre meghatározott költségfüggvény alapján alakul ki a jel útvonala. Az algoritmus „rossz tulajdonsága", hogy az egyszer már elhelyezett jelvezetékeken a továbbiakban nem lehet módosítani.

A Route saját algoritmust is tartalmaz a különleges kezelést igénylő jelek (a tápfeszültség-vonalak, memóriaáramkörök buszvezetékei) behuzalozására.

A teljes kártya huzalozása mellett lehetőségünk van egy kisebb egység (egyetlen jel vagy egy ablakban levő jelek) huzalozására is. Bizonyos területek azonban kitilthatók ebből a folyamatból. A huzalozás egy előre meghatározott hálón, a huzalozási griden történik. A via elhelyezéséhez külön háló definiálható.

A huzalozás során a Design Rule Check program folyamatosan ellenőrzi a terv helyességét. „Óv" minket az egymáshoz túl közel kerülő jelvezetékektől, valamint a kapcsolási rajzzal ellentétes összeköttetés megteremtésétől.

### Feltépve

A PADS-Route — az alkalmazott algoritmus tulajdonságai miatt — sok esetben nem nyújt 100%-os eredményt. Ehhez olyan huzalozó algoritmusra van szükség, amely képes a már lehelyezett jelvezetékek módosítására. Ezzel a jó tulajdonsággal rendelkezik a Ripup & Retry néven ismert algoritmus. Ezen az elven működik a PADS-Superrouter modul.

A Superrouter három fázisban, 100%-os eredménnyel huzalozza be a megtervezett kártyát.

Az első fázis a már ismert Costed Maze algoritmusú huzalozó eljárás, amely a folytonos szabad csatornák kihasználásával 90% feletti huzalozottságot ér el.

A második fázis a Ripup & Retry algoritmus. Ennek lényege, hogy ha a program úgy „ítéli", hogy egy jelet egy már meglevő jelvezeték miatt nem képes bekötni, úgy az útban levő összeköttetést megszünteti, „feltépi", majd más irányban kísérli meg bekötni.

A huzalozás harmadik fázisa az optimalizálás. Ennek során a program minden jelet egyenként vagy kis csoportokban feltép, majd újra beköt. Ez biztosítja a felesleges kerülők és rétegközi átvezetések megszüntetését.

A Superrouter az üres, a részlegesen és a teljesen behuzalozott kártyákon egyaránt használható. A program egyidejűleg 12 jelrétegen végzi a huzalozást. A Maze algoritmus működéséhez újabb költségfüggvényeket definiáltak. A Ripup algoritmus is paraméterezhető: beállítható a feltépett jelek adatait tartalmazó verem mélysége és az iterációk száma.

### Odébbtolva

A Superrouter csak 32 bites változatban létezik (képünkön a már megszűnt 16 bites látható), futtatásához tehát minimálisan 386-os processzorral működő számítógépre és 2-16 Mbájt extended memóriára van szükségünk.

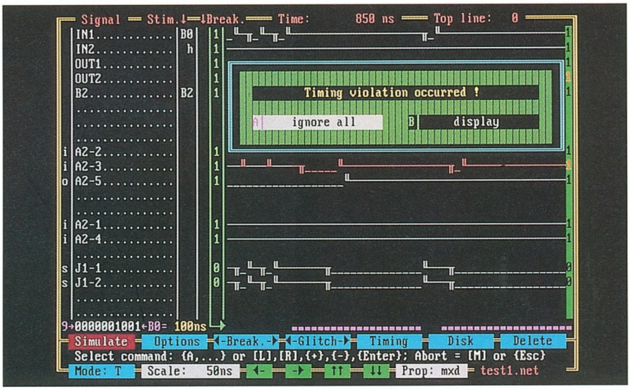
A PADS-Powerrouter modul még tökéletesebb megoldást kínál a kártyák teljesen automatizált huzalozására. A Powerrouter a MAXRoute befejező huzalozó programjának PADS adatbázison, PADS felhasználói környezetben működő változata. Három változatban létezik: 286-os (Windows alatt), valamint 386/486-os PC-n, illetve Sun munkaállomáson működhet.

A Powerrouter — az eddig bemutatott algoritmusoktól eltérő módon — úgynevezett Shove eljárással huzalozza a kártyát. Lényege az, hogy a Ripup algoritmussal szemben a Shove nem tépi fel az akadályozó jelvezetéket, hanem annak szegmenseit — hasonlóan az emberi gondolkodásmódhoz — egyszerűen odébbtolja, biztosítva a szabad utat az új jelnek. Az eljárás eredménye optimalizálás nélkül is jobbnak tekinthető a Ripup algoritmussal huzalozottnál.

A Powerrouter többféle üzemmódban (kézi, autointeraktív módban és 16 jelrétegen) működve automatikusan végzi feladatát. A különleges követelményeket támasztó analóg és nagyfrekvenciás kártyák tervezését is támogatja. A Powerrouter minden korábban ismertetett szolgáltatással képes segíteni az automatikus tervezést. Kétirányú kapcsolata van az ismertebb elektronikus tervezőrendszerekkel.

## Professzionálisan

A PADS-2000 programcsomag teljes körű professzionális tervezőrendszer 386/486-os PC-kre. A korszerű szoftvertechnológiának és a 32 bites architektúrának köszönhetően a program méltó vetélytársa lehet a lényegesen drágább, munkaállomás-alapú rendszereknek is határadataiban, pontosságában (1 mikronos felbontás) és funkciógazdagságában .

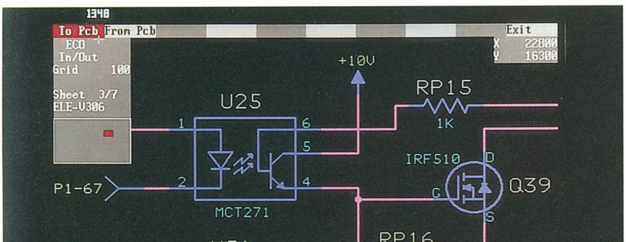



A PADS-2000 tartalmazza a PCB, a Place, a Route, a Plot és a Drill modulok összes szolgáltatását: az analóg kártyák tervezését tetszőleges szögben elforgatható alkatrész-pozicionálással, ívelt vezetékezéssel, automatikus kitöltőfólia-generálással, könnycsepp alakú forrszemgenerálással.

A program igen gyors grafikus felülettel rendelkezik, és támogatja a hardveres scrollt biztosító grafikus kártyák használatát.

Jelenleg a PC-alapú számítógépek körében a PADS-termékek közül a Logic sémaeditor modullal és a Ripup vagy a Push & Shove huzalozó modullal kiegészített PADS-2000 rendszer nyújtja a legjobb teljesítményt.

A Sun SPARCstation-alapú munkaállomásokon a PADS-2000/UX változat futtatható. A program felhasználóit nem korlátozzák tervezési határadatok, igaz, használatához legalább 12 Mbájt RAM és 400 Mbájt lemezterület szükséges.




## Korlátlanul

A PADS-2000+ és PADS-2000/UX+ változatban a hagyományos algoritmusú huzalozó funkciók helyére a teljes kiépítésű Powerroutert építették be.

A PADS-2000/UX egy igazán professzionális elektronikus tervezőrendszer magja, amelyet a PADS-View sémaeditor és a PADS-Forcerouter huzalozó egészíthet ki.

A PADS-View alapváltozatához egy hatékony sémaeditor, egy digitális és analóg szimulátor, illetve egy hullámforma analizátor tartozik. A tervek szerint ez év negyedik negyedévében jelenik meg a PADS-Logic/UX, amely a PADS-View örökébe léphet.

A PADS-Forcerouter forradalmian új huzalozási technikával dolgozó, különlegesen nagy teljesítményű huzalozó modul. A Forcerouter nem használ gridet. A huzalozási képet geometriai alapobjektumokból állítja össze. Ennek az új technikának köszönhetően az adatbázis mérete kb. 10%-a a klasszikus (griden működő huzalozók által használt) adatbázisnak, és lecsökkent a huzalozási idő is. Speciális funkciói támogatják az analóg, a nagyfrekvenciás és a hibrid áramkörök tervezését is. Különleges szolgáltatás a Crosstalk analízis, amely a nagyfrekvenciás jelvezetékek automatikus huzalozását az áthallási szempontok figyelembevételével végzi.

A PADS rendszerrel végzett munka végső fázisa az áramköri lap gyártási

előkészítése. A PADS-PCB modul menürendszerébe beilleszthető PADS-Plot és PADS-Drill szolgál erre a célra.

## Segítséggel

A PADS-Plot a Gerber-kóddal vezérelhető fotoplotterek számára állít elő működtető állományt, illetve lehetővé teszi a Gerber-állomány megjelenítését mátrix- és lézernyomtatón, vagy hagyományos tollplotteren. A PCB által létrehozott rétegrajzok mindegyikéről elkészíthető a Gerber-vezérlőállomány, amelyhez mindig készül egy lista a felhasznált apertúrákról.

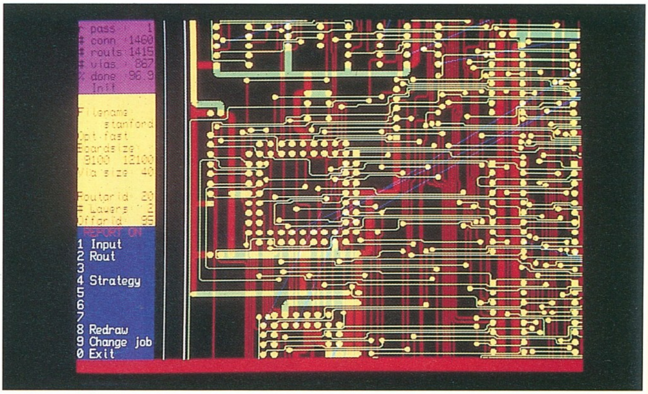
A PADS-Drill modullal az NC fúrógépet vezérlő Excellon formátumú kód készíthető el. Egy állományban tetszőleges számban, adott eltolással szerepelhetnek egy kártya furatai. Így akár egyetlen befogással egy nagyméretű panel összes furata elkészíthető, elegendő ezután feldarabolni a nyomtatott áramköri lapot a kívánt méretre.

A PC-Gerber program Gerber-formátumú fotoplotter-vezérlőállományok interaktív megjelenítését és szerkesztését teszi lehetővé. A PC-Gerber teljesen önálló termék, működtetéséhez nem szükséges más PADS modul megléte. Létezik 386/486 processzorra és Sun munkaállomásra adaptált változata is.

Igen érdekes lehetőséget biztosít a tervező számára a PADS-Thermal modul. Segítségével már a paneltervezés kezdeti szakaszában vizsgálhatók a majdani áramkör hőmérsékleti viszonyai. A PADS-2000 vagy a PADS-PCB modul segítségével kézzel vagy automatikusan elkészített alkatrész-elrendezés analizálja a PADS-Thermal modult. Egy- vagy kétoldalasan szerelt kártyák vizsgálhatók, az eredmény színes grafikus hőtérkép formájában jelenik meg.

A modul figyelembe veszi a kártya helyzetét (gravitáció!), a természetes vagy mesterséges léghűtést, a szomszédos kártyák által termelt hőmennyiség hatását. Amennyiben a kapott eredmény indokolja az alkatrészek átrendezését, úgy az a modulból való kilépés nélkül megtehető, az új eredmény azonnal kiértékelhető. A módosított alkatrészelrendezés természetesen visszatölthető a PADS-2000 vagy a PCB modulba, és a javított termikus paraméterekkel elvégezhető a huzalozás.

A PADS rendszerhez számos konverziós segédprogram is tartozik, amelyekkel a PADS adatbázisok átalakíthatók különféle formátumra (és vissza) úgy, hogy azokat más programokban felhasználhassuk, vagy adatokat cseréljünk közöttük.




A PADS-DXF program segítségével a PADS-2000 vagy a PCB modullal készült kártyageometriák konvertálhatók DXF formátumot használó gépészeti tervezőrendszerek (AutoCAD, EUCLID-IS, ...) számára. Az így kapott tervek alapján hozzákezdhetünk a kártyákat tartalmazó készülékház megtervezéséhez.

A PADS ASCII formátumú adatbázisok a PADS-PCIF programmal P-CAD PDIF formátumra, a CADPADS programmal Redac ASCII formátumra, a PADS-MIF programmal pedig a Mentor tervezőrendszer által használt formátumra alakíthatók oda-vissza.

A PADS-GERBIN program a fotoplotterek által használt Gerber-állományokat értelmezi. A PADS természetesen előállítja a tervezett kártya Gerber-kimenetét, ezzel szemben a GERBIN programmal akár egy ismeretlen rendszerrel készült kártya fotoplotter-vezérlő állománya is PADS formátumra alakítható. A szolgáltatásnak főleg egy régebben készült kártya módosításakor lehet nagy jelentősége.

A PADS-Susie és a PADS-TIM grafikus logikai, illetve időzítésszimulátor programok. Segítségükkel a logikai tervezési hibák jelentős része kiszűrhető a kártya legyártása és tényleges üzembe helyezése előtt. A programok és a hozzájuk tartozó több ezer elemes könyvtárak 16 és 32 bites verzióban egyaránt léteznek.

Mindkét program színes, grafikus idődiagramokat készít az áramkör kiválasztott pontjain megjelenő jelekről, illetve lehetőséget ad a legkülönfélébb vizsgálóminták — az úgynevezett tesztvektorok — interaktív, grafikus előállításához. Mivel egy nagyobb áramkör teljes működésének szimulációja igen hosszú ideig is eltarthat, adott a lehetőség a kapcsolás egy kisebb részletének vizsgálatára is.

## Hazai vizeken is

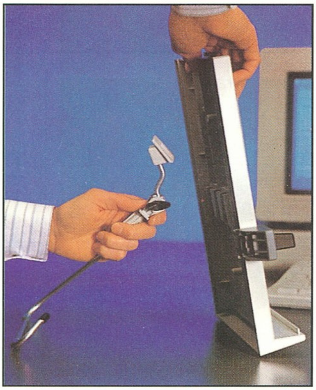
Mint tudjuk, a PADS-termékek magyarországi forgalmazója teljesen díjmentesen rendelkezésre bocsátotta a 16 bites PADS rendszer limitált képességű változatát, a PADS-Basicet. A 3 db HD lemezen szállított Basic rendszer tartalmazza a magyar nyelvű leírást is. A szoftver futtatható minden legalább 286-os processzorral rendelkező PC-kompatibilis számítógépen (EGA vagy VGA videoadapterrel, egérrel és legalább 7 Mbájt szabad lemezterülettel rendelkezik). Segítségével kisebb méretű kártyák gyártásra kész tervezése is elvégezhető.

A Basic-változat igazi haszna elsősorban az, hogy segítségével megismerhetjük a PADS rendszert. A teljes rendszer megvásárlása nélkül, költségmentesen tanulmányozhatók a PADS által nyújtott szolgáltatások, megtanulható a használata.

A PADS tervezőrendszer hazai felhasználóinak jó hír, hogy a közeljövőben megjelenő új verziók minden támogatást megadnak a nemzeti karakterkészletek — így a speciálisan csak a magyar nyelvben szereplő karakterek — használatához is. A forgalmazó CADserver Kft. biztosítja, hogy a felhasználók az angol mellett magyar nyelvű tankönyvekhez is hozzájussanak. Kezdő felhasználók számára nagy segítséget jelent a telefonos „hotline" tanácsadás, amelyet a regisztrált PADS-tulajdonosok mellett a PADS-Basic alkalmazói is igénybe vehetnek.

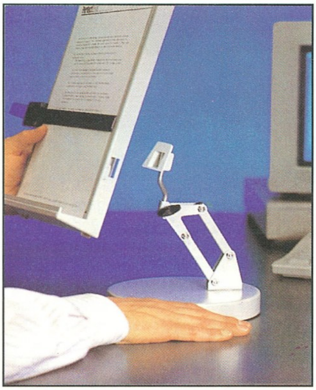
**Lóth Tamás**

PerfectData®

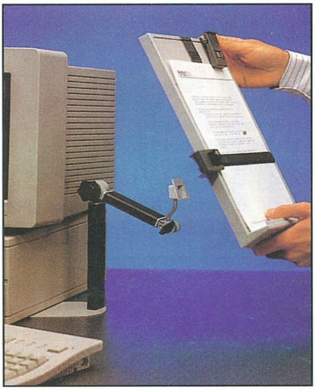


1. **Támasztékos**
Könnyű, könnyedén mozgatható. A legolcsóbb modell. (Ezen nincs digitális óra.)

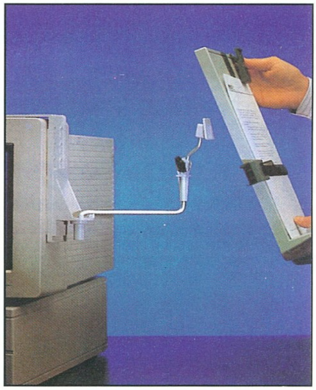
# Kézirattartók — kívánság szerint



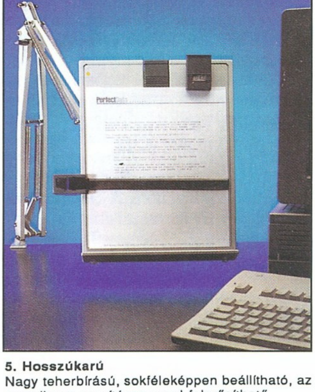
2. **Talpas**
A legnépszerűbb változat. Bármilyen asztalfelületen könnyedén, kis helyen lerakható, máshová bármikor áttehető.



3. **Alátétes**
Kis helyet foglal el. A számítógép, a monitor vagy az írógép alá csúsztatott támaszték hordozza az irattartó táblát.



4. **Felragasztható**
Kétoldalas tapadószalaggal rögzíthető a monitor oldalára. Asztalfelületet nem igényel. Használaton kívül a tábla leemelhető róla.



5. **Hosszúkarú**
Nagy teherbírású, sokféleképpen beállítható, az asztallapra szorítócsavarral felerősíthető.

## Általános szolgáltatások:

— Beépített digitális órával felszerelt papírcsíptető.
— A kéziratok vastagságázoz igazodó, átlátszó sorvezető.
— A nagyobb méretű eredetiket is rögzítő szerkezet.
— Állítható dőlésszögű tábla.

## Az igényesebbeknek néhány extra lehetőség:

— Lábpedálos működtetésű, motorizált sorvezető. Ezáltal a gépelés megszakítása nélkül, folyamatosan követhető az olvasott szöveg.
— Olvasólámpa a táblára erősítve. Különösen a nehezen olvasható kéziratokhoz hasznos.
— Nagyító sorvezető az apróbetűs szövegekhez. Egyszerre két normál sornyi tekinthető át vele.
— Táblatoldalék a szélesebb kéziratokhoz.

Cédrus Karolina Áruház
Budapest XI., Karolina út 17. Tel.: 166-2111 • Fax: 185-2221

INFORMÁCIÓKÉRÉS: 50 ▶

# A CEB-vírusok
# Egy terjedési mechanizmus tanulságai

**Ma már — sajnos — közismert, hogy terjedésük alapján a vírusok két fő csoportra oszthatók: az egyik a bootszektorba települ be, a másik hozzáépül a COM és EXE kiterjesztésű állományokhoz. A bootvírusok jelentik az alattomosabb ellenséget, de a fájlvírusok körében is vannak rejtett veszedelmek.**

Az állományhoz hozzáépülő vírusok elleni védekezés egyszerűbb. Ha ugyanis a fájl mérete növekszik, akkor valószínűleg vírus támadta meg. Ám létezik egy terjedőben lévő olyan fajta is, amely semmiféle változást nem okoz a futtatható fájlokban, és nem ír felül semmit. A trükk, amit alkalmaz, nagyon egyszerű, de ugyanakkor nagyon hatásos. Az operációs rendszer azon tulajdonságát használja ki, hogy ha ugyanazon elnevezésű fájl COM, EXE és BAT kiterjesztéssel egyaránt szerepel, akkor a felsorolás szerinti prioritási sorrendben (C-E-B) előbb szereplő állomány fog elindulni. (Kivéve természetesen, ha a parancssorban a fájlkiterjesztést is megadjuk.) Ha tehát létezik egy PELDA.COM, egy PELDA.EXE és egy PELDA.BAT fájl, akkor a PELDA parancs hatására a PELDA.COM lép működésbe.

Egy ilyen módon fertőző vírus keres egy BAT vagy EXE kiterjesztésű fájlt (esetleg egy COM kiterjesztésűt EXE-re nevez), majd ugyanilyen néven létrehoz egy COM kiterjesztésű állományt, és abba saját magát másolja be. Így amikor a felhasználó a BAT vagy az EXE fájlt akarja futtatni, valójában a vírus indul el. Ha a vírus rejtett állománnyá teszi az általa létrehozott COM fájlt, akkor a lappangási idő alatt a változást sem lehet megfigyelni.

Az ilyen vírus egyszerű futtatható fájl, magas szintű programozási nyelven is létrehozható, s mert terjedelme a rejtőzködés szempontjából nem számít, egészen nagy is lehet, bonyolult rejtőzködési és kártevő algoritmussal.

Az első CEB-vírus valószínűleg az AIDS II volt. 1990 áprilisában fedezték föl. Mérete 8064 bájt, tehát jóval hosszabb a korábbi vírusoknál. Megtalálása azért egyszerűbb, mert az általa létrehozott COM fájlokat nem teszi rejtetté. Különösebb kárt nem okoz, mindössze zenél egy kicsit, és óvszer használatára hívja fel a figyelmet, hogy elkerüljük a legközelebbi fertőzést.

Erről a vírusfajtáról a vírusirtó társaságok hallgatnak, vagy egy kézlegyintéssel elintézik a dolgot. A kézlegyintés annak szól, hogy elég könnyű a vírustól megszabadulni: mindössze a pluszban létrehozott COM fájlt kell letörölni, a vírus felderítésére pedig a hagyományos szekvenciakereső eljárások ugyanúgy használhatók. (Mondják ők.)

A hallgatás oka pedig az, hogy ezek a vírusok természetüknél fogva sokkal hosszabbak. Létezik már 29 kilobájt hosszúságú is! Egy átlagos vírus hossza kb. 2 kB. A legegyszerűbb CEB-vírus 8-10 kB-nál kezdődik, így visszafejtése legalább 4-5-ször, de akár 10-15-ször annyi munkát is igényelhet. (Ha valaki azt hiszi, hogy egy C-ben vagy Pascalban megírt program visszafejtése könnyebb, az nagyot téved.) Ráadásul a többletmunka nem is éri meg, mert a CEB-vírusok többsége egyszerűen terjed, s a COM fájl törlésével valóban kiirtható. Ennek viszont megvan az a veszélye, hogy a vírus törlésével értékes adatok is kárba vesznek. Képzeljenek el egy vírust, amely találomra összecserélgeti a merevlemez szektorait. Amíg aktív, addig minden cserét visszacsinál, viszont amint kiirtják, teljes lesz az összevisszaság. Ilyen manipulációra példa a dBase vírus. Ez a DBF fájlokat manipulálja, és a változásokat egy BUG.DAT fájlban tárolja, amelynek letörlésével a DBF állományok már visszaállíthatatlanok.

Mivel a CEB-vírusokat többnyire C vagy Pascal nyelven írják, ezért a lefordított kódra már csak a fordító „stílusa" jellemző, nem pedig a vírus írójáé. A szekvenciakeresés alapja, hogy egy vírus apróbb részletei alapján is könnyen azonosítható, és nem keverhető össze más programmal. Ám a C-ben vagy Pascalban írt vírusok rövid részlete könnyen összetéveszthető bármelyik olyan programmal, amelyet ugyanazzal a fordítóval készítettek. Hosszabb szekvenciákra, más felismerési mechanizmusokra van tehát szükség.

A következő probléma az ilyen vírusok elleni védekezés megoldásakor tűnik elő. Azt még ellenőrizni lehet, hogy COM fájl létrehozásakor létezik-e az adott néven EXE vagy BAT fájl, de mi van akkor, ha mondjuk a vírus először egy XXX.CON fájlba másolja magát, majd ezt nevezi át a REN XXX.C?N NEV.??M utasítással. Létrejön egy NEV.COM fájl egy NEV.EXE mellett, ám a vírusvédelem ilyen esetben elvileg sem tudja eldönteni, hogy az átnevezéskor történik-e fertőzés. Ki lehet próbálni. Egyetlenegy általam ismert vírusvédelmi program sem figyelmeztetett ilyenkor a fertőzés veszélyére!

Végül maradt a legegyszerűbb megoldás. Végig kell nézni az összes fájlt, hogy azonos alkönyvtárban létezik-e közöttük azonos néven COM, EXE vagy BAT kiterjesztésű. Ha ilyeneket találunk, akkor azok akár CEB-vírusok is lehetnek. Mindössze néhány szoftvernél találtam ilyen egybeesést. Ezek között azonban olyanok voltak, mint az MS-DOS 5.00 Shell programja, vagy például a Deskview. Így mindössze akkor gyanakodhatunk CEB-vírusra, ha több ilyen gyanús fájl létezik, és ezek hossza esetleg meg is egyezik. Az Alaplap mostani számának lemezmellékletén közreadjuk a CEBVIR.EXE néven található egyszerű programot, amely a neve után megadható alkönyvtárban végignézi az összes fájlt, és ha azonos néven COM, EXE vagy BAT fájlt talál, akkor ezeket a hosszukkal együtt kiírja. Mivel a jövőben a CEB-vírusok elterjedése is várható, és a szekvenciakeresők (például a SCAN) nem tudnak majd lépést tartani velük, ezért célszerű a CEBVIR programot is felvenni az otthoni víruskeresők listájába és időnként ellenőrizni vele a merevlemezt.

Ne feledjük: nemcsak a fájlhossz-növekedés utalhat fertőzésre, hanem a COM állományok szaporodása is!

**Hornák Zoltán**

## Csináld magad!

# Hogyan ismerhetjük fel a vírusokat?

**A felhasználók jelentős része beszerez néhány vírusirtót, és azok többé-kevésbé rendszeres használatával védekezik. Mit tehet azonban, ha vírusra utaló jelenségeket tapasztal, de a vírusirtó programok semmit nem jeleznek? Ebben az esetben sem kell kétségbeesni. Az alábbi néhány tanács segíthet megoldani a problémát, feltételezve persze az átlagosnál valamivel nagyobb jártasságot, rendszerismeretet, és a DEBUG programok használatát.**

Legfontosabb a vírus megtalálása, és néhány tulajdonságának megállapítása. Elsőként talán azt kell kideríteni, hogy beül-e a vírus rezidensen a memóriába vagy sem. A rezidens vírusok között igen veszélyesek és nehezen felfedezhetők is lehetnek. A memóriában megbújó, a folyamatokat ellenőrző vírus bizonyos esetekben (így például víruskereséskor) beavatkozhat, és hamis adatokkal megtévesztheti a vírusirtókat. Példa a rejtőzködésre a magyarok közül a Töltögető (amely az eredeti bootszektort mutatja) és a PIF-PAF vírus (amely az eredeti 21h megszakítási címet adja vissza).

Veszélyesek azért lehetnek a rezidens vírusok, mert nagyon gyorsan terjedhetnek, különösen, ha sok fájlműveletet végzünk. E két tulajdonságot talán a V2000 vírus esetében figyelhetjük meg a legjobban. A V2000-et ugyanis valaki beépítette egy régi, védelemmel nem rendelkező SCAN programba. A SCAN indításakor a V2000 rezidenssé vált, majd a SCAN működése során a vizsgált állományokba beépült.

A rezidens vírusok jelenlétének megállapításához kihasználhatjuk, hogy amikor az a memóriában maradva figyeli az eseményeket, legtöbbször átállítja a lemezkezelésben fontos szereppel bíró 13h vagy 21h megszakítót. E két interrupt megvizsgálásával a rezidens vírusok jelentős része megfogható. Rendszerindítás után indítsuk el a DEBUG programot. A DEBUG bejelentkezése után írjuk be a D 0:0 FF parancsot. Ekkor az alábbi keretes részben láthatóhoz hasonló kiírást kapunk.

Most tekintsük meg a 0000:0080-as sor 4—7. bájtját. A sorrend megfordításával és a bájtok párba állításával megkapjuk a 21h interrupt címét. Jelen esetben ez 0D36:0D17. Ha a 21h interrupt 9xxx:xxxx címre mutat, akkor fájlvírusunk van, amely a rendszerindítás során használt állományokat (COMMAND.COM stb...) vagy az előbb behívott DEBUG-ot fertőzte meg. A 13h megszakítón „lógó" vírusokat már nehezebb megtalálni, mert ezek rendszerint bootvírusok, amelyek az operációs rendszer betöltődése előtt „lopták el" az interruptot. Annyit lehet tenni, hogy a DEBUG-ba az

```
A
 INT 13
 (Enter)
```

begépelése után a T parancsot ismételgetve figyeljük, hogy mielőtt A000:0000-nál nagyobb címre érünk (rendszerint C800-as címre), nem kerül-e át a vezérlés 9xxx-es címre. Ha igen, akkor bizony bootvírusunk van. Másik módszer, amely kevesebb vírus esetén hatásos, azt használja ki, hogy sok vírus tartalmaz kódolatlan szöveget. E szövegeket a memóriában megtalálva gyanakodhatunk az adott vírus jelenlétére. Ha például a Stoned vírus rezidens, akkor a DEBUG D 9000:F800 FFFF parancsára a szöveges oszlopban megtalálhatjuk a „Your PC is now STONED!" szöveget.

Jól kihasználhatjuk azt is, hogy a vírusoknak olyan memóriára van szükségük, amelyet az operációs rendszer nem használ. A bootvírusok ennek érdekében a memória tetejére épülnek be (ezért ésszerű a Stoned-ot a 9000:F800 helyen keresni), majd ezt a területet az operációs rendszer számára zárolják. Ennek megtörténtét a DEBUG D 0:413 parancsával észlelhetjük. Itt alapesetben a 80,02 bájtok állnak (280h=640). Ha ez megváltozott, akkor valószínűleg egy bootvírussal vagyunk gazdagabbak.

A fájlvírusok a memóriafoglalást más módszerrel oldják meg: DOS funkcióval, vagy az MCB manipulálásával. Az előbbieket egy megfelelő memóriavizsgálóval (jó a Norton SI) vehetjük észre. Ha a rezidens (TSR) programok között észreveszünk olyat, amelyiknek nem volna szabad rezidensnek lennie, és ráadásul még a 21h vagy 13h megszakítást is elveszi, akkor valószínűsíthető egy fájlvírus jelenléte. Az MCB

```
0000:0000  E8 56 2B 02 56 07 70 00-16 00 F8 0A 56 07 70 00   .V+.V.p.....V.p.
0000:0010  56 07 70 00 54 FF 00 F0-23 FF 00 F0 23 FF 00 F0   V.p.T...#...#...
0000:0020  7B 08 36 0D 0C 0A 36 0D-23 FF 00 F0 23 FF 00 F0   {.6...6.#...#...
0000:0030  23 FF 00 F0 B1 03 F8 0A-39 04 F8 0A 56 07 70 00   #.......9...V.p.
0000:0040  78 0F 36 0D 4D F8 00 F0-41 F8 00 F0 98 09 36 0D   x.6.M...A.....6.
0000:0050  39 E7 00 F0 59 F8 00 F0-1A 01 CE 26 D2 EF 00 F0   9...Y......&....
0000:0060  23 FF 00 F0 E0 18 70 00-6E FE 00 F0 50 07 70 00   #.....p.n...P.p.
0000:0070  54 08 36 0D A4 F0 00 F0-22 05 00 00 C0 2C 00 C0   T.6....."....,..
0000:0080  3F 14 2B 02 17 0D 36 0D-F4 02 6C 29 2F 03 6C 29   ?.+...6...l)/.l)
0000:0090  BC 02 6C 29 4E 0D 36 0D-64 0D 36 0D 66 63 2B 02   ..l)N.6.d.6.fc+.
0000:00A0  73 09 36 0D 9C 06 70 00-45 14 2B 02 45 14 2B 02   s.6...p.E.+.E.+.
0000:00B0  45 14 2B 02 45 14 2B 02-81 02 C6 0B 0C 18 70 00   E.+.E.+.......p.
0000:00C0  EA 46 14 2B 02 14 2B 02-45 14 2B 02 45 14 2B 02   .F.+..+.E.+.E.+.
0000:00D0  45 14 2B 02 45 14 2B 02-45 14 2B 02 45 14 2B 02   E.+.E.+.E.+.E.+.
0000:00E0  45 14 2B 02 45 14 2B 02-45 14 2B 02 45 14 2B 02   E.+.E.+.E.+.E.+.
0000:00F0  45 14 2B 02 45 14 2B 02-45 14 2B 02 45 14 2B 02   E.+.E.+.E.+.E.+.
```

Például a Victor vírus így fedezhető fel:

```
DEBUG
-n gyanus.com
-l
-s cs:100 ffff '*.*'
135F:C3AA
Ha most megvizsgáljuk a '*.*' helyét:
-d 135f:c390 c3cf
135F:C390   00 F4 7A 2C 00 C6 D1 A1-A0 9F 44 22 5C 42 5E 5D   ..z,......D"\B^]
135F:C3A0   E8 B1 00 E9 1E BF 4A 00-21 00 2A 2E 2A 00 43 4F   ......J.!.*.*.CO
135F:C3B0   4D 45 58 45 03 3F 3F 3F-3F 3F 3F 3F 3F 3F 3F 3F   MEXE.???????????
135F:C3C0   E3 05 00 21 00 00 00 00-00 20 4A 00 21 00 95 C2   ...!.....J.!...
```

Tovább keresve a fájlban, megtalálhatjuk a Victorra jellemző szöveget:

```
135F:CC90   0D 0B 0D 00 00 0D 0A 56-69 63 74 6F 72 20 56 31   ....... VictorV1
135F:CCA0   2E 30 20 54 68 65 20 49-6E 63 72 65 64 69 62 6C   .0 The Incredibl
135F:CCB0   65 20 48 69 67 68 20 50-65 72 66 6F 72 6D 61 6E   e High Performan
135F:CCC0   63 65 20 56 69 72 75 73-0D 0A 45 6E 68 61 6E 63   ce Virus..Enhanc
135F:CCD0   65 64 20 76 65 72 73 69-6F 6E 73 20 61 76 61 69   ed versions avai
135F:CCE0   6C 61 62 6C 65 20 73 6F-6F 6E 2E 0D 0A 54 68 69   lable soon...Thi
135F:CCF0   73 20 70 72 6F 67 72 61-6D 20 77 61 73 20 69 6D   s program was im
135F:CD00   70 6F 72 74 65 64 20 66-72 6F 6D 20 55 53 53 52   ported from USSR
135F:CD10   2E 0D 0A 54 68 61 6E 6B-73 20 74 6F 20 49 76 61   ...Thanks to Iva
135F:CD20   6E                                                n
```

manipulálását észrevehetjük az AFD programmal (a DEBUG ide nem jó).

```
AFD
 A
  MOV  AX,CS
  DEC  AX
  MOV  DS,AX
  ADD  AX,[0003]
  INT  20
  (CTRL+HOME)
 D 100
 Négyszer kell az F1-et
megnyomni.
```

Ezután az AX regisztert megvizsgálva 9FFF-et kell kapnunk, ellenkező esetben egy fájlvírus van a memória tetején. A nem rezidens vírusokat ezekkel a módszerekkel nem lehet észrevenni, hiszen nem tartózkodnak huzamosabb ideig a memóriában.

A fájlvírusok felismerésének elősegítésére a vírusokra jellemző hossznövekedéseket a vírustáblázatok közlik, ami alapján többé-kevésbé azonosítható is a vírus. A fájlhosszt nem növelő fájlvírusok egy részét nagyon profi vírusírók készítették (ezért egyszerű eszközökkel nem lehet a vírust észrevenni).

Egyre gyakororibb azonban az a primítív és galád vírusírási módszer, hogy a vírussal egyszerűen felülíratják a megfertőzendő állományokat. (Pontosabban többnyire azok felületének egy részét.) Mivel fájlvírusból több mint ezer létezik, felismerésükhöz elég nehéz jól használható tanácsot adni. Az egyszerűbb (nem önkódoló) vírusok jelentős részében megtalálható a *.*, *.com, *.exe vagy a *.exe,*.com jelsorozat. Ezt a DEBUG-ban a következő utasítással lehet megkerestetni:

S CS:100 FFFF 'keresendő szöveg'

A bootvírusok a lemez kijelölt helyén, a bootszektorban találhatók. (Nem mindegyik, hiszen léteznek olyanok, amelyek az eredeti, fertőzés előtti bootszektort adják vissza!)

Ha önmagukat nem álcázó bootvírussal van dolgunk, akkor azt hamar felfedezhetjük, hiszen a boot (illetve merevlemezen a masterboot) megváltozik.

Érdemes tehát a boot-ot és a masterboot-ot egy fájlba még vírusmentesen menteni, hogy fertőzés esetén az eredeti állapot visszaállítható legyen. Ez különösen hasznos akkor, ha a szektort a vírusok eltüntetik. Például a Michelangelo-Stoned páros a masterboot-ot egyszerűen „megeszi". Ennek az lesz a következménye, hogy winchesterről nem lehet rendszert indítani.

**Rákos Péter**

# Újdonságok McAfee 93-as csomagjában

A széles körben használt szoftverekhez feltétlenül szükség lenne magyar nyelvű leírásra. Legjobb példa erre a McAfee-féle víruskereső és vírusirtó programcsomag, amelyet szinte mindenütt használnak, de angol nyelvű használati utasításait — és azok folyamatosan módosuló részeit — tapasztalataink szerint viszonylag kevesen tanulmányozzák át. Ennek hiányát segítünk az alábbiakban egy kicsit pótolni.

Nem sokkal az igen nagy publicitást kapott és március 6-án tetőzött Michelangelo-hullám után újabb közellenség tűnt fel. Érthetetlen rendszerösszeomlások történtek. A bűnös a K vírus, ez a 4928 bájt hosszúságú „állatka" volt. A Scan program 86-osnál korábbi verziói még nem ismerték, a vírus korrekt eltávolítását pedig a későbbi kiadások sem oldották meg, s a fertőzött állományt a CLEAN program ugyanúgy felülírja és törli, mitha a SCAN-t a /D kapcsolóval használtuk volna.

A ViruScan, NetScan és CleanUp június 24-i keltezésű, 8.6V93 jelzésű verziójában a keresőprogram néhány igen hasznos kiegészítéssel bővült. A /HISTORY kapcsolóval a megadott naplóállományt folyamatosan vezeti, és nem írja azt felül, ellentétben a /REPORT kapcsoló használatával.

Újdonság az /AF, a /CF és az /RF kapcsoló is, melyekhez szervesen hozzátartozik a CleanUp /GRF kapcsolója. A „SCAN C: /AF [fájlnév]" parancs a C: meghajtó állományait úgy immunizálja, hogy nem az állományok végére biggyeszti az ellenőrző-visszaállító kódokat, hanem a kapcsoló mellett megadott állományban gyűjti azokat.

Végre ezt is megértük!

Most már írásvédett vagy önellenőrző programokat is lehet immunizálni, akár ismeretlen fájlvírusok ellen is. A /CF kapcsoló az ellenőrzést, az /RF pedig az „immunkód" eltávolítását szolgálja.

Az utolsó lényeges bővülés a batchalkalmazásokat érinti. A 73-as verziótól kezdve a ViruScan és a NetScan az eddigi három helyett már ötféle értéket adhat az ERRORLEVEL változónak (0-1-2-3-4).

Ennek köszönhetően külön kezelhetjük a billentyűzetről történő programmegszakítást is, ami által rugalmasabb alkalmazásokra nyílik lehetőség.

**Nagy Gábor**

# Ami a Vírushatározóból kimaradt...

Az Alaplap Könyvek sorozatban végre megjelent Vírushatározó a McAfee nómenklatúrát tekinti támpontnak, és annak alapján közli a vírusok összefoglaló táblázatát is. Miután azonban a könyv nyomdába adásakor még csak a 89-es verzió volt forgalomban, itt most közreadjuk a később felbukkant vírusok főbb jellemzőit összefoglaló listát.

Kihagytuk azonban a Generic család C, D, E, H, J, M, N, Q, R, S, V1, V2, V3, V4 jelzésű tagjait, mert azokról egyelőre a nevükön kívül egyebet nem közöltek.

## JELMAGYARÁZAT

Méretnövekedés:
Vált. = Változó hosszúságú
N/A = A vírus nem fűződik hozzá az állományhoz
Felülír = A vírus felülírja a fájl elejét, a fájlméret nem változik
Számértékek = A bájtokban megadott méretnövekedés, amennyivel a fájl nagyobb lesz, ha megfertőződik
A károkozás módja:
B = Megrongálja vagy felülírja a bootszektort
D = Megrongálja az adatállományokat
F = Részben vagy teljesen újraformázza, illetve felülírja a lemezt
L = Közvetve vagy közvetlenül megrongálja a fájlkapcsolatokat
O = Az operációs rendszer futtatási műveleteit befolyásolja
P = Tönkreteszi a programállományokat és az átfedő (overlay) fájlokat
Jelölések:
x = Igen
. = Nem

| MEGNEVEZÉS | | MÓDSZER | | | FERTŐZÉS HELYE | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Név | Rövidített névkód | Lopakodó technika | Önmagát titkosítja | Memóriába installálódik | COMMAND.COM | .COM állományok | .EXE állományok | Overlay állományok | Floppy bootszektora | Merevlemez bootszektora | Partíciós tábla | A fertőzött program méretnövekedése | KÁROKOZÁS MÓDJA |
| A-403 | [A-403] | . | . | x | x | x | . | . | . | . | . | Felülír | L |
| AIDS | [N1] | . | . | . | . | x | x | . | . | . | . | Felülír | P O L |
| AIDS II | [A2] | . | . | . | . | x | x | . | . | . | . | ? | O |
| Beware | [Bwr] | . | . | . | . | x | . | . | . | . | . | 442 | P |
| Brainy | [Bry] | . | . | x | x | x | . | . | . | . | . | 768 | O |
| Busted | [Bst] | . | . | . | x | x | . | . | . | . | . | Felülír | O P L |
| C | [CV] | . | . | . | . | x | x | . | . | . | . | N/A | P |
| Catman | [Ctm] | . | . | x | . | . | . | . | x | . | x | N/A | O |
| CD-10 | [D2] | x | x | x | . | . | . | . | . | . | . | N/A | L |
| CkSum | [Cks] | . | . | x | x | x | . | . | . | . | . | 1233 | P O |
| Criminal | [Crm] | . | . | x | x | x | . | . | . | . | . | 2615 | P |
| DM-B | [Dmb] | . | . | x | x | x | . | . | . | . | . | 400 | P |
| Europe-92 | [E92] | . | . | x | x | x | . | . | . | . | . | 728 | O L D |
| FGT | [GN] | . | . | . | x | x | . | . | . | . | . | 651 | P |
| Fil | [Fil] | . | . | x | x | x | . | . | . | . | . | 1658 | P D |
| Flu-2 | [Fl2] | . | x | x | x | x | x | . | . | . | . | 2112 | P |
| Francois | [Fra] | . | . | x | . | . | . | . | x | . | x | N/A | O |
| Free | [Free] | . | . | . | . | x | . | . | . | . | . | 692 | P |
| Generic File | [GenF] | . | . | x | ? | x | ? | ? | . | . | . | Változó | P |
| Ghost Dos-62 | [Gho] | . | . | . | . | x | . | . | x | x | . | 2351 | B P |
| Gliss | [Gls] | . | . | . | . | x | . | . | . | . | . | 1247 | P |
| Gomb | [Gomb] | . | x | x | x | x | . | . | . | . | . | 4091 | P |
| H-2 | [H-2] | . | . | x | x | x | x | x | . | . | . | 1962 | O |
| Hafen | [Hafn] | . | . | . | . | . | x | . | . | . | . | 809 | P |
| Here | [Hre] | .. | . | . | . | x | . | . | . | . | . | N/A | O |
| Idle | [Idle] | . | . | x | . | x | x | . | . | . | . | 2332 | P O |
| Ill | [Ill] | . | x | . | x | x | . | . | . | . | . | 1016 | P |
| JD | [JD] | . | . | x | x | x | . | . | . | . | . | 158 | P |
| KBug | [Kbu] | . | . | x | . | x | x | . | . | . | . | 1598 | P O |
| Kilroy | [Klr] | . | . | x | . | . | . | . | x | . | x | N/A | O |
| KU-448 | [KU] | . | . | . | x | x | . | . | . | . | . | 448 | L |
| Kuzmitch | [Kzm] | . | . | . | . | x | . | . | . | . | . | 1064 | P O |
| LZ | [LZ] | . | . | . | . | . | . | . | . | . | . | Felülír | O |
| M-128 | [M128] | . | . | x | x | x | . | . | . | . | . | 128 | L |
| Nazi | [Ram] | . | . | . | . | x | . | . | . | . | . | N/A | O P |
| Null | [NL] | . | . | . | x | x | . | . | . | . | . | 733 | P |
| P-45 | [P45] | . | . | . | . | x | . | . | . | . | . | N/A | P |
| Parasite-2B | [Ps2] | . | . | . | x | x | . | . | . | . | . | 903 | P O F |
| Peach | [Pch] | . | . | x | x | x | x | . | . | . | . | 889 | P |
| Polish-583 | [P583] | . | . | . | x | x | . | . | . | . | . | 583 | P P D |
| Queen's | [GenB] | . | . | x | . | . | . | . | x | . | x | N/A | O B |
| Sadist | [Sadt] | . | . | . | . | . | x | . | . | . | . | 1436 | P O |
| Saratoga | [Doodle] | . | . | x | . | . | x | . | . | . | . | 656 | P L B |
| Shield | [Shd] | . | . | x | . | x | . | . | . | . | . | 127 | O |
| Suriv B | [Surivb] | . | . | x | . | x | x | x | . | . | . | 1813 | P O |
| Unk | [Unk] | . | . | x | x | x | x | . | . | . | . | 1015 | P |
| V-Label | [Label] | . | . | x | x | x | x | . | . | . | . | Felülír | L P |
| V914 | [914] | . | . | . | . | x | . | . | . | . | . | 914 | O P |
| VHP-2 | [VHP2] | . | . | . | . | x | . | . | . | . | . | N/A | P O |
| Windmill | [Wm] | x | . | x | . | . | . | . | x | . | . | N/A | B |
| 382 (2 változat) | [Pir] | . | . | . | x | x | x | . | . | . | . | Felülír | L O P |
| 654 | [640] | . | . | . | . | x | . | . | . | . | . | 654 | P |
| 1030 | [G4] | . | . | x | . | . | x | x | . | . | . | 1030 | O P L |
| 1452 | [1452] | . | . | x | x | x | . | . | . | . | . | 1452 | O P |
| 1530 | [1530] | . | . | . | . | x | x | . | . | . | . | 1530 | P O |
| 2560 | [2560] | x | . | x | . | x | x | . | . | . | . | 2560 | O P |

**A Mikrobazár rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk.**

**A kereskedelmi tevékenységet szolgáló apróhirdetések tarifája gépelt soronként (60 karakterenként) 300 Ft. Kérjük, hogy a hirdetés díját a Cédrus Kiadó Kft-nek az Általános Értékforgalmi Banknál vezetett 204-19417 számú számlájára utalják át, vagy postautalványon fizessék be a Cédrus Kiadó Kft címére (1441 Budapest VIII., Reguly Antal u. 8.), a hátoldalon feltüntetve, hogy apróhirdetés. A befizetést igazoló szelvényt a közlendő hirdetési szöveggel együtt az Alaplap szerkesztőségéhez (a kiadóéval azonos címre) küldjék el.**

**A nem kereskedelmi célú egyéni hirdetések közlése INGYENES!**

## ELAD

**Enterprise programok** eladók. Válaszboríték ellenében listát küldök. 2000 program, sok kedvezmény, ajándék. Cím: Zemen László, 1164 Budapest XVI., Olló u. 16.

Eladó **TVC**-hez C-64 hangkártya, 9 Epromos cartridge, 1541-es floppymeghajtó. Ugyanitt eladó C-64-hez Final Cartridge III. Cím: Vámos György, 8008 Székesfehérvár, Pf. 1.

Eladó **XT számítógép:** 20 MB HDD, 360 kB FDD, 640 kB RAM, multi I/O kártya, CGA kártya, monokróm papírfehér monitor. Ugyanitt cserélnék AT játék- és felhasználói programokat. Cím: Zalavári Miklós, 9023 Győr, Ipar u. 100.

Eladó **ST 238-R-es Seagate winchester,** 32 MB, tele felhasználói programokkal. Ára: 15 000 Ft. Cím: Szabó Zsolt, 8630 Balatonboglár, Gyöngyvirág u. 26.

Olcsón eladó **monokróm (Hercules) és színes SVGA monitor**. Cím: Martincsek Balázs, tel.: 149-5600.

Eladó PC-hez 2M típusú, Covox kompatibilis, mono- és sztereo **hangkábel** szoftverekkel és dokumentációval. Cím: Antalóczy Tibor, 6800 Hódmezővásárhely, Vas u. 5. Tel.: (62) 45-431.

**PC video-vezérlőkártyák** (Hercules, CGA, EGA, VGA) hardveres ékezetesítése CWI vagy egyéb tetszőleges kódkészlet szerint. PC Turbo klubtagoknak kedvezmény! Cím: Lóth Tamás, 1558 Budapest, Pf. 76.

**Számítástechnikai oktatás** IBM PC gépen bármilyen témában a legolcsóbban! Beszerzési tanácsadást és programkészítést is vállalok! PC Turbo klubtagoknak kedvezmény! Cím: Fridl György, 1116 Budapest XI., Szalóki u. 6.

Programokra, segítségre, cseretársakra van szükséged? Az ASIS megoldja problémáidat! Bárhol laksz, bármilyen géped van, írj! Kérésre ingyenes tájékoztatót küldünk. Cím: ASIS, 1425 Budapest, Pf. 729. Tel.: 142-8075.

**Ingyen adok PC-programokat!** Válaszboríték ellenében listát küldök. Cím: Besse Marianna, 4032 Debrecen, Egyetem sugárút 67. I./3.

## VESZ

Egyetemi hallgató kér olyan segítséget, hogy juttassák el címére **fölösleges vagy üzemképtelen hardver eszközeiket** (részegységeiket). Cím: Dömsödi Mihály, 2038 Sóskút, Orgona u. 11.

## CSERÉL

Elcserélném **Atari Portfolio** gépemet 386-os AT bővítésére alkalmas 4x1 MB-os RAM-ra vagy egyéb műszaki árura. Ugyanitt eladó 4 MB-os Amiga RAM bővítő, amely 1,5 MB-ig kiépített. Cím: Major Nándor, 9300 Csorna, Kossuth L. u. 53.

Keresem a Microsoft **Macro Assembler 6.0**-ás változatát, de az 5.1 vagy e fölötti verzió is érdekel. Keresem továbbá a Borland C++, a Microsoft Quick C és a Turbo Pascal 6.0 programokat. Csereajánlataim az alábbiak: MS Windows 3.1, Norton Desktop for Windows, MS Word for Windows 2.0, MS Excel 3.0 for Windows, MS Works 2.0 for Windows, Superbase IV. for Windows, HDC First Apps for Windows, Chessmaster for Windows. Cím: Lukácsy Gergely, 1118 Budapest XI., Törökugrató u. 4. I./1. Tel.: 173-5044.

**IBM PC játék- és felhasználói programokat** cserélek Hercules monitorra. Cím: Józan László, 2344 Dömsöd, Ságvári u. 20.

## Norton, Peter:
## Az IBM PC és a PS/2 anatómiája.

Peter Norton alapkönyvek.
(Budapest — London, 1992.
Novotrade Kiadó kft. — Prentice Hall, 340 oldal. Ára: 899,- Ft.)

Ismerje meg Peter Norton irányításával az IBM PC és a PS/2 felépítését, programozását!

Peter Nortonnak ez az immár harmadik — alapvetően átdolgozott és korszerűsített — kiadást megért könyve méltán népszerű az olvasók körében, hiszen kiadásról kiadásra nyomon követi és bemutatja az IBM személyi számítógépek fejlődését. (A korábbi magyar kiadások *Fedezzük fel az IBM-PC-t!* címmel a Műszaki Könyvkiadónál jelentek meg.)

Hasznos és könnyen megérthető tudnivalók tárházát kínálja e könyv mindazoknak, akik nemcsak passzívan akarják használni személyi számítógépüket, hanem arra is kíváncsiak, hogy mi és hogyan működik a rendszerben, és adott esetben be is kívánnak avatkozni működésébe. Az érdeklődő átfogó ismeretet szerez a számítógépek belső felépítéséről, fő egységeinek működéséről és a kapcsolódó legfontosabb perifériákról. Megismerkedhet a legelterjedtebb operációs rendszerekkel, áttekintést kaphat a közkedvelt programozási nyelvekről és a számítógép-programok felépítéséről.

A témaköröket rövid, szellemes, gyorsan beírható és kipróbálható példaprogramok illusztrálják.

INFORMÁCIÓKÉRÉS: 51 ▶





INFORMÁCIÓKÉRÉS: 53 ▶

## „Ha a lehetetlent lehetne letetni...”

# Sziget a nyelvek óceánjában

Mostani számunkkal újraindítjuk Kaleidoszkóp sorozatunkat mindazok számára, akik eszük rugalmasságát a potenciális alkalmazások területén is ki akarják próbálni — akiknek a számítástechnika jóval többet jelent, mint számítást és technikát.

1768-ban III. György angol király kezdeményezésére a Királyi Tudományos Társaság tudományos expedíciót szervezett a csendes-óceáni szigetvilágba. A csillagászok szerint ugyanis a következő évben Tahiti szigetéről lehet majd legjobban megfigyelni a Vénusz elhaladását a Nap koronája előtt, ebből pedig pontosabban ki lehet számítani a Nap—Föld távolságot.

Az angolok tudományos lelkesedésében persze a franciákkal vívott nyílt háborúskodás lezárultával az a lázas igyekezet is szerepet játszott, hogy ahol csak lehet, megelőzzék vetélytársaikat a még jórészt ismeretlen világtenger meghódításában.

Az expedíciót vezető Cook kapitány és csapata meglepetéssel tapasztalta, hogy polinéziai kísérőjük, akit Tahiti szigetén vettek a hajóra, gondtalanul társalgott az új-zélandi maorikkal. Még nagyobb volt az ámulatuk az indonéz szigetvilágban, ahol kísérőjük az indonéziai malájokkal is olyan jól meg tudta magát értetni, hogy még tolmácsolni is tudott.

A polinéz probléma azóta sem hagyja nyugodni a tudós társadalmat. Honnan jöttek? Mikor szállták meg a szigetvilágot? Hogy őrizték meg ősi nyelvüket és kultúrájukat a távoli szigeteken, akkora távolságokra egymástól?

Mert a távolságok elképesztőek. Képzeljünk el a Föld felszínén egy gömbháromszöget, amelynek egyik oldala akkora, mint a Budapest—Tokió távolság (ez feleltethető meg a Hawaii-szigetek — Húsvét-sziget távolságnak), a másik akkora, mint innen Dél-Afrika



INFORMÁCIÓKÉRÉS: 55 ▶



INFORMÁCIÓKÉRÉS: 56 ▶

(ilyen messze van a Hawaii-szigetektől Új-Zéland), a harmadik pedig feleljen meg a Budapest—Washington távolságnak (ennyire van Új-Zéland a Húsvét-szigettől). Bármilyen hihetetlen, egy ilyen méretű háromszögön belül lényegében azonos kultúrájú, alapjaiban megegyező nyelvű népek laknak. Arról már ne is beszéljünk, mekkora területet kapnánk, ha az indonéz szigetvilágot is hozzászámítanánk.

Mindez azonban még mind semmi. Az igazi meglepetés akkor következett be, mikor bebizonyosodott, hogy Indonéziától nyugatra, a Föld másik felén (majdnem akkora távolságra, mint innen Új Zéland) rokon nyelvű és rokon kultúrájú népek lakják Madagaszkárt is, a világ negyedik legnagyobb szigetét.

A madagaszkári malgasz nyelvet az ausztronéz (Humboldttól származó régebbi elnevezése szerint: maláj-polinéz) nyelvcsaládnak nem a polinéz, hanem az indonéz ágából származtatják. A nyelvek közötti eltérésben azonban ez távolról sem jelent olyan félelmetesen nagy különbséget, mint az általunk ismert nyelvcsaládoknál megszokhattuk. Hogy csak egyetlen példát mondjunk: Madagaszkár szigetén így számolnak 1-től 10-ig: 1 — isza, 2 — roa, 3 — telo, 4 — efatra, 5 — dimy, 6 — enina, 7 — fito, 8 — valo, 9 — szivy, 10 — folo, az egyik polinéz nyelven pedig így: 1 — taszi, 2 — lua, 3 — tolu, 4 — fa, 5 — lima, 6 — ono, 7 — fitu, 8 — valu, 9 — hiva, 10 — nafulu.

Mostani feladatunkat egy pici sziget nyelvéből vettük, amely valójában már kívül is esik a fenti óriás háromszögön. Új-Guineához közel, a Karolina szigetekhez tartozó Nukuoro-korallzátonyon beszélik ezt a polinéz nyelvjárást, amely így a legnyugatibb polinéz nyelvjárásnak tekinthető.

Megadunk nyolc szót nukuoro nyelven, majd ezek jelentését, de összevissza kevert sorrendben. A feladat egyszerű: meg kell állapítani, hogy mit jelent másik nyolc nukuoro szó, aminek már a jelentését sem adjuk meg. Egyszerű. Nem?

## Nukuoro szavak

1. hakakaina
2. hakaunu
3. hakonohonoho
4. hanohano
5. heiununa
6. nnoho
7. nohonoho
8. uunu

## Magyar jelentéseik

a) etetés
b) itatott
c) ittak
d) járkált
e) szomjúság
f) üldögélt
g) ültek
h) ültetgetett

## További nukuoro szavak

9. hakakai
10. hakakaikai
11. heikai
12. heikaina
13. kai
14. kaikai
15. kaina
16. kkai

A kérdésekbe bújtatott feladatok tehát a következők (mindenki annyit végezhet el belőlük, amennyit tud):

1. Hogyan feleltethetők meg egymásnak az A és a B halmaz elemei? Hogy jött rá?

2. Mit jelentenek a C halmazbeli szavak? Miért?

3. Milyen elemi tevékenységek elvégzéséről kellene gondoskodnia annak a programnak, amely a fenti nukuoro szavakat elő tudná állítani?

4. Mi lehetne e program inputja?

5. Milyen további nukuoro szavakat tudna képezni a fenti szavak elemeiből?

6. Mit gondol, miről kapták a nevüket a Társaság-szigetek? És miért kérdezzük ezt éppen a fent elmondottakkal kapcsolatban?

7. Milyen ötletei vannak a verseny győzteseinek jutalmazására?

A verseny 1993. áprilisi feladványunkkal ér véget, előző Kaleidoszkóp versenyünkhöz hasonlóan minden szám után létraversenyszerűen összesítjük a szerzett pontokat. Előzetes tervünk szerint 1993 júliusában lesz az eredményhirdetés. A „létra" felső fokaira kerülteket díjazzuk, de hogy hány jutalmazott lesz, az attól is függ, milyen aktivitással vesznek részt olvasóink a versenyben, továbbá hogy milyen minőségű megfejtéseket küldenek be. (Ha még emlékeznek rá: múltkori versenyünkben olyan kiváló megoldásokat kaptunk, hogy előzetes ígéretünket túlteljesítve több jutalmat osztottunk ki.)

A feladatok tematikájához kapcsolódva módot szeretnénk nyújtani az olvasók egyéni aktivitásának kifejtésére is. Rövid programokat, rutinokat szívesen közlünk mágneslemez mellékletünkön, bőséges kommentárokkal.

Felhívjuk a figyelmet Programozástechnika rovatunkra, ahol mostani számunkkal kezdjük el a SNOBOL nyelv ismertetését. Ez a nyelv kiváló eszközt jelenthet nyelvi feladatok viszonylag könnyű megoldására (ráadásul sok más nem aritmetikai probléma elegáns megoldására is). Természetesen sok feladat más nyelvekben is ügyesen megoldható — kérjük Olvasóinkat, hogy hívják föl erre a mi figyelmünket is, és rajtunk keresztül többi olvasóinkkal is osszák meg ismereteiket.

Mostani feladatunk megoldásának beküldési határideje: 1992. október 10. A megoldásokat a szerkesztőség címére kérjük beküldeni: Alaplap szerkesztősége, Kaleidoszkóp, 1441 Bp., Pf. 74.

**Vargha Dénes**

## SZÁMÍTÁS- ÉS IRODATECHNIKAI KELLÉKEK AZ IMPORTŐRTŐL

★ Mágneslemezek
★ Streamer kazetták
★ Mágnesszalagok
★ Optikai lemezek
★ Festékkazetták
★ Tonerek
★ Leporellók, etikettek
★ Másolópapírok

3% kedvezmény!

## VISZONTELADÓK, FIGYELEM!

200 különféle festékkazetta printerekhez, pénztárgépekhez, továbbá írógépszalagok állandóan raktárról. Ha már máshol nem találta meg a kazettát, amit vevője a ritka géptípushoz kért, forduljon hozzánk!

CORWELL
TRADING & CONSULTING

1143 Budapest, Utász u. 5. • Tel.: 252-43-59

megamicro

SZÁMÍTÁSTECHNIKAI, FEJLESZTŐ, GYÁRTÓ, SZOLGÁLTATÓ RÉSZVÉNYTÁRSASÁG

1145 BUDAPEST, RONA U. 127/B,
TEL./FAX: 252-1500, 183-0378, 252-0341,
251-6911 TELEX: 22-3153

## COMPFAIR-kínálatunkból:

- Számítógép-hálózatok
- Ipari és telepi számítógépek
- Műholdas helymeghatározás – telekommunikáció
- Vezeték nélküli adatátvitel
- Biztonságtechnikai, vagyonvédelmi eszközök
- Autó- és lakásriasztók
- Telefonközpontok, rádiótelefonok
- Aktív memóriakártyás beléptető rendszer

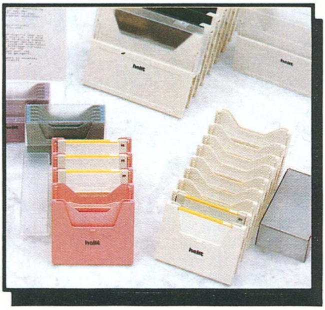
TERVEZÉS – TELEPÍTÉS – ÜZEMELTETÉS

*Keresse kiállításunkat az „A" pavilon 213. sz. standján!*

## A-TÓL AZ Ω-IG, A-TÓL Z-IG, AZ ÍRÓSZERTŐL AZ IRODÁIG!

## Vállaljuk:

Kis- és nagyvállalkozások ütemezett ellátását irodaszerekkel, reprezentatív ajándéktárgyakkal, igény szerinti emblémázással!



***Ismerje meg választékunkat és kedvező szolgáltatási feltételeinket!***

*Budapest VI., Podmaniczky utca 27. * 1067*
*Levélcím: Pf. 71. 1922*
*Tel.: 132-8168 Tel./Fax: 132-0188*

*Iroda Kultúra Szalon*
*Budapest V., Kálmán Imre utca 14. * 1054*
*Tel./Fax: 153-4898*

*Pécs, Szalay Antal utca 12/a. * 7622*
*Tel./Fax: (72) 21-181*

IQ STUDIO

ÍRODA KULTÚRA STÚDIÓ

PRAX

3525 Miskolc, Déryné u. 18.
Tel./Fax.: (46) 347-898,
Kazinczy u. 19. Tel.: (46) 349-619
Eger, Csiky S. u. 17. Tel./Fax: (01) 06 106
Levélcím: 3501 Miskolc, Pf.: 398

Magyarországon a

# TIRIS

A Texas Instruments rádiófrekvenciás azonosítórendszerének magyarországi forgalmazója a

COMPUTER PRAXIS KFT.
MISKOLC

INFORMÁCIÓKÉRÉS: 61 ►

# Milyen VGA rendszert válasszunk?

Az IBM 1987 áprilisában mutatta be a VGA-t. Akkor az volt a személyi számítógépek világában a „menő" videorendszer, alig 4 év alatt azonban már az új számítógépek „kötelező gyakorlatává" vált, mert a hangsúly áttevődött a grafikára. A kiadványszerkesztéshez és a Windows-alkalmazásokhoz — de a játékok túlnyomó többségének vonzó megjelenítéséhez is —, legalább EGA-felbontás kell.

Ugyanakkor a VGA számos probléma forrásává is vált, mert a gyártók saját Super VGA-változataikat kínálják, s megjelent az XGA is, de azt — és a hozzá való monitort — egyszerű földi halandók még nem engedhetik meg maguknak. Akkor hát milyen VGA beszerzésére vállalkozzunk?

A VGA (video graphics array = videografikai tömb) néhány másik, kisebb teljesítményű szabványból fejlődött ki: az MDA-ból (monochrome display adapter), a Hercules grafikai kontrollerből, a 640x200 képpont felbontású CGA-ból (color graphics adapter) és az EGA-ból (enhanced graphics adapter), amely 640x350 képpontjával a szöveget igen jól, és a grafikát is elég szépen adja vissza, de nem teljesen kompatibilis a CGA-val, tervezési hézagjai pedig a szoftverírást elég fárasztó aprómunkává teszik.

Megjelenésével a VGA nem hozott olyan forradalmi változást, mint például a „poloskázás" (bug fix). Csak kevéssel jobb felbontású, mint az EGA (640x480), viszont kiküszöböli annak tervezési hibáit és inkompatibilitását a CGA-val, továbbá sokkal jobb színmegjelenítést tesz lehetővé. Egyidejűleg az IBM kihozta a még nagyobb felbontású, 1024x768-as 8514/A videorendszert is, de az nem lett olyan népszerű, mert elég költséges, és a képernyője interlaced módban dolgozik (interlaced = váltott soros letapogatású), ezáltal nehéz olvasni. (Az interlace trükk lehetővé teszi, hogy egy monitor nagyobb képfelbontást jelenítsen meg, mint amekkorára tulajdonképpen képes, aminek az a mellékhatása, hogy a kép vibrál, fejfájást és látási zavarokat okozva.)

Ha sok pont van a képernyőn, a videokártyának is több memóriára van szüksége azok megjelenítéséhez. Függ a memóraigény a színek számától is. Például számos VGA 320x200-as felbontásban 256 színt tud megjeleníteni, 640x480 képpont esetén azonban már csak 16-ot. Ennek nincs köze a monitor fizikai korlátaihoz, de még a videokártyához sem — kivéve, hogy mennyi memória van ráépítve. A videokártyák memóriaigénye a felbontás és a színek függvényében az alábbi:

| | | |
|---|---|---|
| 640x480 képpont, | 16 szín — | 256 K |
| 640x480 képpont, | 256 szín — | 512 K |
| 800x600 képpont, | 16 szín — | 256 K |
| 800x600 képpont, | 256 szín — | 512 K |
| 1024x768 képpont, | 16 szín — | 512 K |
| 1024x768 képpont, | 256 szín — | 1024 K |

Nem sokkal a VGA megjelenése után a gyártók máris elkezdték kínálni a 800x600-as saját Super VGA-kat. Először azok kártyái is interlaced rendszerűek voltak, mert az akkor rendelkezésre álló monitorok ezt a felbontást másként még nem tudták kezelni. Azóta azonban a helyzet lényegesen megváltozott. A hirdetésekben látható, hogy a VGA-kártyák többségéhez valamiféle Super VGA üzemmódot is ajánlanak, sokszor ingyenes ráadásként. Ez úgy tűnik, mitha nagyon jól járnánk vele, de van benne három csapda.

— Ahhoz, hogy használni is lehessen a Super VGA módot, kétszer annyi pént kell kiadni a megfelelő monitorra, mint




egy normál változat esetén. Tegyük fel, hogy vesz valaki kb. 90 dollárért egy névtelen VGA-kártyát. A dokumentációból kiderül, hogy van 800x600-as, Super VGA módja, 16 színnel. Csak az a baj, hogy a normál VGA-monitor (250–350 dollár) azt nem jeleníti meg, ezért multifrekvenciás monitort kell venni (450–1000 dollár). Az „ingyen" Super VGA ára a kettő közötti különbség.

— A szoftverek közül nem mindegyik tudja kezelni a Super VGA működésmódot, és nincs rá garancia, hogy a videokártya készítője a következő években mindig elérhető lesz, vagy hogy a szoftverekhez mindig elkészíti a szükséges illesztéseket, meghajtószoftvereket. A névtelen gyártótól eredő kártya Super VGA módjából pedig rövidesen „fehér holló" lehet. A szoftverkompatibilitás azonban részben megoldódik, ha néhány gyártó szabványosítási kezdeményezését (VESA = Video Electronics Standards Association) a többi „kártyagyáros" is támogatja, mert akkor a szabványt a szoftverkészítők eleve figyelembe veszik.

— Ha meg is van minden szükséges kellék a magasabb felbontáshoz, azokat a processzornak fel is kell dolgoznia, emiatt a gép általában lassabban működik. A személyi számítógépekben használt kártyák ma még általában elég „süketek", és a munka zömét a központi processzor végzi el. Ezen csak a grafikai koprocesszorok elterjedése tud majd változtatni, az ilyen kártyák azonban ma még elég drágák, 1000 dollár felettiek.

Ahhoz, hogy szemünk nyugodt képet észleljen a monitoron, másodpercenként legalább 60-szor kell a képernyő tartalmát felfrissíteni. Ennél alacsonyabb frekvencia már villódzó vagy vibráló hatást kelt. Néhány Super VGA-kártya képváltási sebessége 70 vagy 72 Hz, ami tehát lényegesen kellemesebb hatású. A baj csak az, hogy ezek nem tudnak együttműködni a papírfehér monitorokkal, s egy sokkal drágább multifrekvenciás képernyő szükséges hozzájuk. Ha

viszont valaki egyébként is ilyen monitort vásárol, akkor már érdemes hozzá 70 vagy 72 Hz-es kártyát választani.

A monitorok 1986-ig vízszintesen és függőlegesen is rögzített pásztázási gyakorisággal működtek. A vízszintes frekvencia a sorok számának (a nem láthatókat is beleértve) és a másodpercenkénti képernyőfrissítésnek a szorzata, a függőleges frekvencia pedig maga a másodpercenkénti képváltás. Ennek alapján a CGA-monitorok frekvenciája vízszintesen 15,75 kHz, függőlegesen 60 Hz. Az EGA-monitorok kártyájának már két normát kell teljesítenie: a 21,8 kHz/60 Hz-es EGA-t, és az említett CGA-t. A szabványos VGA ehhez még hozzáadja a saját 31,5 kHz/60 Hz-es módozatát.

Új megoldást jelentett 1986-tól a NEC Multisync monitora, amely észleli, és önmagával összhangba hozza a kártyák vízszintes frekvenciáját 15 és 31,5 kHz között, függőleges frekvenciáját pedig 50 és 70 Hz között, tehát a monitor ezeken a tartományokon belül bármelyik videokártyával összekapcsolható. Ennek mintájára más gyártók is forgalomba hozták Multisync-szerű monitoraikat, változó frekvenciájú monitor (VFM) elnevezéssel. A NEC már nem árulja az eredeti Multisyncet, csak a továbbfejlesztett változatokat, melyek közül a legnagyobb teljesítményűnek a paramétere 30–66 kHz/50–90 Hz. A konkurencia termékei közül pedig legerősebb a Sony 1304-es típus.

Nem kell megfeledkezni a monitor mérete és a képpont mérete (dot pitch) közötti összefüggésről sem. Minél kisebb ez az érték, annál jobb felbontásban jeleníthető meg a kép, a nagyobb monitorokon pedig a méret nagyobb is lehet az élesség romlása nélkül. A VGA monitorokhoz 4 képpontméret tartozik: 0,34 mm, 0,31 mm, 0,28 mm és 0,26 mm. El kell kerülni a 12"-os képernyőn a 0,34 mm-est, de az a 14"-oson már elfogadható lehet. (Vásárlás előtt célszerű megtekinteni.) A 12"-os normál VGA-hoz is jó a 0,31 mm-es, de a 800x600-as Super VGA-hoz 0,28 mm kell, az 1024x768-ashoz pedig 0,26 mm.

Végül is mit lehet leszűrni ennyi információból? Választani kell mindenekelőtt a normál VGA és a Super VGA között, de mindkét esetben a 16 bites kártyák közül.

Normál VGA esetén könnyű a döntés: rögzített frekvenciájú VGA-monitor (pl. Gold Star, Samtron stb.), névtelen VGA-kártya 256 K memóriával és 16 bites síncsatlakozóval.

Super VGA esetén célszerűbb márkás kártyát választani, mert a főbb szoftverek illesztéseihez így biztosan hozzájuthatunk. Super VGA-kártyához viszont VFM-monitort kell vásárolni, mert nagyon kevés a rögzített frekvenciájú Super VGA-monitor, bár a szabványosodás piacot teremthet számukra.

Kerüljük el a hanyatlóban lévő, ezért esetleg olcsó, de nem igazán előnyös monitorokat. Ezek vízszintes frekvenciája 35,5 kHz alatti, nem érik el a 70 Hz-es képfrissítést, és csak 56 kHz-es interlaced módban tudják megjeleníteni a 800x600 képpontot. Marad tehát két másik, a 800x600-as (pl. NEC 3D vagy a Seiko) és az 1024x768-as csoport. Ez utóbbi „non-interlaced" VFM-ek vízszintes szinkronizációja 50 kHz-ig vagy azon túl is terjed, függőlegesen az 50–87 Hz közötti sávban vannak. (A 87 azért bűvös szám, mert ez az, amit a 8514-es kártya használna, ha nem lenne interlaced módban, és néhány gyártó így is kínálja.) Ha erre a kategóriára van szükségünk, legjobb a Sony 1304-es monitort, vagy ahhoz hasonlót választani, de arra készüljünk fel, hogy nem olcsó mulatság.

Ne felejtsük el, hogy nagy felbontás esetén a monitor kerül sokba, nem pedig a kártya! Ezért lesz az XGA nagyon drága akár néhány éven át is. Ellentétben a merevlemezzel és az alaplappal, a monitor nem új találmány, a katódsugárcsöves technológia már hosszú ideje itt van, és nem lehet újabb rohamos árcsökkenésre számítani. **(-fp-)**

**(Compute, 1991/11)**

# Gyónnak a szoftverkalózok

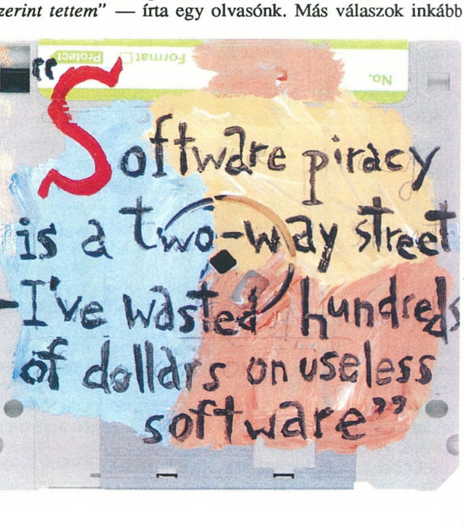
*„Szoftverkalóz vagyok? Juhé! A PC-men lévő összes szoftver illegális, kivéve a nyilvános segédprogramokat. Több mint 800 népszerű szoftverrel rendelkezem. Ezt a különös dokumentumot kalóz módon szerzett DOS 5.0, Windows 3.0, Bitstream Facelift fontok, CorelDraw 2.0 és PageMaker felhasználásával bocsátom rendelkezésükre."* Így kezdődött egy magát „Pir-o-maniac"-nak nevező zsivány levele, aki leírta és elküldte nekünk gondolatait a kalózkodásról.

Több mint 900-an mondták el, hogy készítenek-e és használnak-e illegális szoftvermásolatokat, hogy főleg milyen programokkal kalózkodnak, és milyen körülmények között tartják helyénvalónak a szoftvermásolást.

A legtöbben készítenek és használnak illegális szoftvermásolatot. A válaszadóknak csupán 8 százaléka mondta azt, hogy soha nem készített, és nem is használt illegális másolatot. 79 százalékuk készített, és 87 százalékuk használt már bizonyos esetekben ilyeneket. A válaszadók több mint fele — 61 százalékuk — tette hozzá, hogy az általuk éppen használt szoftverek némelyike illegális.

Ezeknél a statisztikáknál sokkal érdekesebb az, hogy milyen érvekkel támasztják alá álláspontjukat. Sok válaszadó volt bizalmatlan vagy önbíráskodó: *„Ha még nem vették volna észre, a szoftvergyártók előre betervezett elavulást építenek be termékeikbe, hogy a számítógép-használókból minden dollárt „kifejjenek". Ha megvásárolok egy programot, akkor az már az enyém, és természetesen az összes számítógépemre felrakom. Keresem az indokot ahhoz, hogy országos méretű lázadást vagy bojkottot kezdeményezzek, és ez lehetne az."*

Másoknak bűntudatuk van és mentegetődznek: *„Igen, használtam már illegális másolatot. A munkáltatóm akarata szerint tettem"* — írta egy olvasónk. Más válaszok inkább



# A NÉV

## a nagy nevek mögött



KAO - EZ A NÉV ÖNNEK LEHET, HOGY NEM ISMERŐS, DE ISMERŐS OLYAN CÉGEKNEK MINT PL. MICROSOFT, LOTUS ÉS SOK SOFTWARE - GYÁRTÓNAK, AKIKNEK A SAJÁT NEVÜK SZEREPEL KAO LEMEZEKEN. A MÁGNESLEMEZEK JÓ MINŐSÉGE ÉS MEGBÍZHATÓSÁGA A KAO-T AZ ÉSZAK-AMERIKAI PIAC EGYIK LEGNAGYOBB SZÁLLÍTÓJÁVÁ TETTE. TEHÁT, HA A LEMEZEK, STREAMER KAZETTÁK ÉS DAT-OK MINŐSÉGE ÉS MEGBÍZHATÓSÁGA ÖNNEK FONTOS, AKKOR CSAK EGY NEVET TARTHAT SZEM ELŐTT : KAO.

INFORMÁCIÓKÉRÉS: 45 ▼

**KAO®**

MAKROTREND - 1143 Budapest, Hungária krt. 65 - 67. Tel: 183-4356 Fax: 163-7888

filozofikus jellegűek: *„A könyvtáraknak talán díjat kellene felszámolniuk a könyvek kölcsönzésekor, hogy jogdíjat fizessenek a szerzőnek? Sokan éveket töltenek tanulmányokkal, és keményen dolgoznak egy-egy könyv megírásán, hasonlóan, mint a programozók."*

A megjegyzések világossá teszik, hogy a válaszadók becsapva érzik magukat a szoftverek indokolatlanul magas árai, a szigorú felhasználási feltételek, vagy a reklámban felfújt és nem teljesített ígéretek miatt, ezért elégtételt vesznek maguknak azzal, hogy megfosztják a szoftvergyártókat bevételük bizonyos hányadától.

Egy olvasó ezt így összegezte: *„Amikor kemény pénzeket adunk ki használhatatlannak bizonyuló vackokért, feljogosítva érezzük magunkat, hogy valamilyen jó árut kalózkodással elorozva szerezzük vissza belőle valamit."* Az illegálisan másolók többsége (87%) arra hivatkozik, hogy szüksége van olyan példányra, amelynek alapján előzetesen meggyőződhet a szoftver használhatóságáról. Sokan hozzáteszik, hogy ha tényleg szükségük van rá, akkor végül is fizetnek érte. Ahogy egy válaszadó írta: *„Miután eldöntöttem, hogy az adott programot használni akarom, akkor megveszem, mert hozzá akarok jutni a kézikönyvekhez, a további fejlesztésekhez és a technikai szolgáltatásokhoz."* Sokan keserűen emlékeztettek arra, hogy mennyi pénzt költöttek már a reklámban ígérteket nem teljesítő programokra. *„A szoftverkalózkodás kétirányú utca — mondta egy válaszadó. — Sok száz dollárt pazaroltam el a használhatatlan szoftverekre. Hányszor csapták már be vevőiket a a szoftvergyártók abszurd állításokkal reklámozott vagy koncepciótlan és gyenge termékeikkel!"*

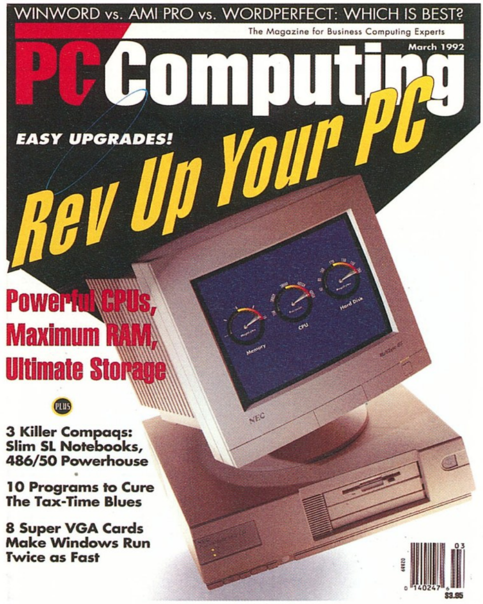
Mások azzal érveltek, hogy a kipróbálásra történő másolás tulajdonképpen hasznos az ipar számára. *„Már 6000 dollár érték felett vásároltam a vállalatnak szoftvert az általam használt kalózmásolatokkal szerzett tapasztalatok alapján."* Egy másik vélemény: *„Néhány barátommal közös pénzalapot hoztunk létre, hogy meg tudjunk vásárolni olyan termékeket, amelyeket külön-külön egyikünk sem engedhetne meg magának. Nem jogszerű? Igaz, de így legalább megvettünk egy példányt, aminek alternatívája a nulla lett volna."* Volt, aki szerint *„a kipróbálásra készült másolat a legjobb reklám."*

A kisvállakozások gyakran elkerülhetetlennek tartják, hogy kalózszoftvert használjanak, mert nem engedhetik meg maguknak azok megvásárlását. Ugyanakkor ezzel a spórolással többnyire számítógépes felszereltségüket tudják javítani, ami végül majd mégis a legális szoftverforgalom malmára hajtja a vizet. A kalózkodás ügyében sokan különbséget tesznek a munkahelyi és az otthoni PC-k között. *„Az irodai PC-men lévő szoftverek legtöbbje legális, az otthoni PC-men lévők legtöbbje másolt* — írta egy olvasónk. — Úgy érzem, hogy ez így teljesen rendben van, mert az otthoni szoftverekkel nem folytatok kereső tevékenységet."

## A kalózmásolatok slágerlistája

(A PC Computing körkérdése alapján)

| | |
|---|---|
| WordPerfect | 34% |
| Lotus 1-2-3 | 22% |
| dBASE | 15% |
| Norton Utilities | 14% |
| PC Tools | 13% |
| Windows | 11% |
| Harvard Graphics | 9% |
| AutoCAD | 7% |
| Excel | 6% |
| MS-DOS 5.0 | 5% |




Másoknál is előjött az a gondolat, hogy a „házi használatú" kalózmásolatokból az iparnak inkább haszna van. *„A szoftvergyártóknak bátorítaniuk kellene, hogy a munkahelyi szoftverekről az otthoni PC-kre másolatot készítsenek. Kinek van arra ideje munkaidőben, hogy az oktatóprogramokat lefuttassa, és a nagy szoftvercsomagok minden jellemző tulajdonságát kifürkéssze? A programhasználatban való gyakorlottság és a makrók, apró fogások ismeretéből fakadó nagyobb termelékenység növeli magának a szoftvernek az értékét is. A személyes befektetés fejleszti tovább egy megbízható régi termék használhatóságát, amikor a versenytársak arra csábítanak, hogy próbáljuk ki az újdonságot. A szoftvergyártóknak fel kellene ismerniük, hogy ahol a munkát hazaviszik, ott ahhoz a szoftverhez való ragaszkodás erősödik."*

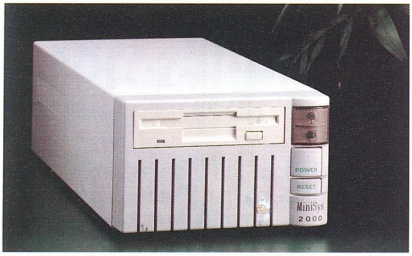
Sok válaszadó hasonlítja a szoftvert a könyvhöz vagy a CD-hez, néhányan pedig úgy gondolják, hogy a jogi keretek igencsak elavultak. *„A szoftverhasználat 'engedélyezése' nevetséges elképzelés* — írta egy válaszoló. *Elképesztően cinikus dolog a gyártók részéről, hogy megpróbálják a szoftvert annyiszorosan megfizettetni, ahány gépen azt a vásárló használja. Akkor még talán volt ennek az elképzelésnek létjogosultsága, amikor egy számítógép nagyon sokba került, és a szoftverek szerzőinek évente tíz csomag értékesítéséből kellett biztosítaniuk megélhetésüket. Csakhogy az idők megváltoztak!"*

A leggyakrabban visszaköszönő reflexió az volt, hogy a magas árak nem önmagukban jelentik a szoftverkalózkodás egyetlen okát. *„Az ár csupán kis része az indoknak* — írta egy olvasó. — *Ha a szoftvergyártók a bejegyzett felhasználóknak szuperminőségű technikai támogatást, olcsó továbbfejlesztett verziókat, és más valóságos előnyöket kínálnának, akkor sokkal többen vennék meg a terméket."* **(-ft-)**

**(PC Computing, 1992/3)**

INFORMÁCIÓKÉRÉS: 64 ▼

# MiniSys




286-os, 386-os mini számítógépek
tetszőleges kiépítettségben.

KOMDEX számítógépek (USA),
NOTEBOOK-ok, alkatrészek.

**C. REX Kft.**
1015 Budapest I., Szabó Ilonka u. 79.
Tel./Fax: 201-5010
Bolt: XII., Alkotás u. 13.
(Bejárat: Greguss u.)

# MODISOFT

## IBM XT/AT-kompatibilis BSC szinkron PC-kártya.

Adatátvitel telefonvonalon, telexvonalon
és rádióhullámokon át minden mennyiségben.

## V23-as modemek.

Egyedi tervezésű
adatátviteli berendezések, rendszerek.

**MODISOFT KFT.**
**1476 Bp. Pf. 61.**
**1093 Bp. IX., Üllői út 155.**
**Tel.: 157-4497**

INFORMÁCIÓKÉRÉS: 65 ▼

INFORMÁCIÓKÉRÉS: 74 ▼

# Újdonságaink!




COMPUTERBOOKS
1126
BUDAPEST, TARTSAY VILMOS U. 12.
TEL.: 1751 564, 1753 591
FAX.: 1753 591

KERSZÖV computer

Kerszöv Kft.
1027 Budapest II., Bem rkp. 51.
Tel.: 115-9605, 115-8498
Fax: 115-8498

| | Végfelhasználói ár | Viszonteladói ár |
|---|---|---|
| GEA laptop GLT-216A (286-12)<br>1 MB RAM, 40 MB HD, 1,44 MB FD, bővítő kártyahely | 112 500– | 99 900,– |
| GEA unibook SX3U1 (386SX 20)<br>2 MB RAM, 40 MB HD, 1,44 MB FD, Fax-modem-szkenner kártyahely | 136 000,– | 129 900,– |
| HUN 386 DX 33 MHz notebook<br>4 MB RAM, 120 MB HD, 1,44 MB FD | 179 900,– | 172 400,– |
| Bicom B240i mini notebook (286-16)<br>2 MB RAM, 40 MB HD, menedzserkalkulátor-funkciók, 1 kg | 137 500,– | 125 000,– |




| | 1–5 db esetén | 6–10 db esetén | 10 db felett |
|---|---|---|---|
| AT babyház 200 W tápegységgel | 4 820,– | 4 690,– | 4 525,– |
| Minitorony 200 W tápegységgel | 6790,– | 6 560,– | 6 380,– |
| Színes SVGA monitor | 23500,– | 22 900,– | 22 100,– |

A fenti árak az ÁFA-t nem tartalmazzák.

## OOP — a Borland C++ 2.0 „színpadán"

# Új sorozatunk

Időszerű témába merülhet az érdeklődő olvasó, ha mostantól keresi is ezt az Alaplap hasábjain: minden hónapban egy írás az objektumorientált programozással foglalkozik majd. Az OOP bemutatására a Borland szoftverház C++ 2.0 rendszerét választottuk. Ez a nyelv a szabványos OOP és C++ kiterjesztéseken kívül más újdonságokat is tartogat számunkra — ahol lehetséges (és lényeges), ott ezekre is kitérünk.

Legelőször is arról essék néhány szó, hogy mi is az a C++. Mint a nevéből is következtetni lehet rá, ez a C programozási nyelvi kiterjesztése. A hagyományos C napjainkban ünnepli harmincadik születésnapját, így időszerűnek nevezhető, hogy utódja született. Mivel minden gyermeknek két szülője van, ez a C++ esetében sem lehet másképp. Az apa (aki nem adta nevét a porontyhoz) a Simula. A kisded az apjától csak az OOP szervezésmód támogatását örökölte. Minden más tulajdonsága a mamától való, de itt is jelentős evolúciós fejlődést mutat az őséhez képest. A gyermek Amerikában született, az AT&T óriáscég egyik szülőszobájában. A szülést Bjarne Stroustrup vezette. Az anya továbbra is kitűnő egészségnek örvend.

Mint már kiderült, a C++ az OOP technikát az ANSI C kiterjesztéseként valósítja meg. Ugyanakkor OOP technikát már a Pascal nyelv egyes implementációi is támogatnak, de ezek terminológiája és nyelvi struktúrája annyira eltér a C++ nyelvéétől, hogy közös tárgyalásuk nem célravezető.

Először az objektumorientált programozás alapjait kell egy kicsit körüljárnunk. Ez a programozási stílus a megfelelő fordítóprogramon kívül egy speciális látásmódot is kíván a használójától, s ehhez néhány dolgot tisztázni kell, mielőtt a konkrét lehetőségekről és módszerekről szó esnék.

Az első fogalom, amivel meg kell ismerkednünk, az osztály. Ez az ANSI C struktúrákhoz és unionokhoz hasonlóan származtatott típus. Tulajdonképpen a C struktúrák kiterjesztése. Habár ebből következik, de fontos kiemelni, hogy egy osztálydefiníció (pontosan mint a struktúradefiníció) nem jár helyfoglalással. A definiált osztály neve típusdefiníciót takar. A későbbiekben ilyen típusú változókat, valamint az ilyen változókra mutató pointereket deklarálhatunk. A struktúra- vagy osztálydefinícióban megadjuk a tagok neveit, valamint típusait, a deklarációban pedig létrehozzuk a definícióban megadott jellemzőjű változókat. Az „osztály típusú" változókat objektumnak nevezzük. A struktúrákkal való kompatibilitás következménye, hogy osztálydeklarációt a struct kulcsszóval is jelölhetünk.

A legfeltűnőbb különbség az, hogy a C++ struktúrák az adatokon kívül függvényeket is tartalmazhatnak. Ezeket nemes egyszerűséggel függvénymezőnek — vagy Borland-terminológiával member functionnek — nevezhetjük. Ez a tulajdonság lehetővé teszi, hogy az egy egységben tárolt adatokat, és az ezeken manipuláló függvényeket szorosan összezárjuk. Ezáltal a program világosabb, jobban olvasható lesz. Van itt egy első hallásra nem túl szembetűnő következmény. Gondoljuk csak meg, ha egy ANSI C struktúrataghoz akarunk hozzáférni, annak a neve elé oda kell írni, hogy melyik konkrétan deklarált struktúrából vegye ki a fordító. Ez egyértelmű, mivel a definíció nem foglalt helyet egyetlen struktúra számára sem, a struktúratag csak akkor él, ha létrehozunk (deklarálunk) egy adott típusú változót. Ugyanez lesz a helyzet a C++ függvénymezőkkel is. Aktivizálni őket kizárólag egy létező objektumváltozón keresztül lehet. Egy olyan változón keresztül, amelynek típusa a funkciót tartalmazó osztály. Talán egy kicsit furcsa a következő programblokk:

```
struct mystruc myobj;
myobj.classfunc(par);
```

Itt a mystruc egy osztály, amelyet régebben definiáltunk. Ez az osztály rendelkezik egy int classfunc(int) kinézetű függvénymezővel. (A két int típusdefinícó helyébe bármilyen más típusnevet írhattunk volna). A második utasítás a függvénymezőt a myobj nevű objektumon keresztül aktivizálja. A myfunc nem is férhet egykönnyen hozzá más objektumpéldányok vátozóihoz, csak azéhoz, amelyiken keresztül aktiválva lett. Létező objektum nélkül pedig (minő fájdalom) nem is aktiválhatjuk. Az előző példánál maradva a pucér

```
classfunc(par)
```

függvényhívás ellen a fordítóprogram hevesen tiltakozni fog.

Az osztályokat (csakúgy, mint a struktúrákat) egymásba ágyazhatjuk. A beágyazott osztályt őstípusnak, amelyikbe beágyazzuk, azt pedig származ-

### Definíció — deklaráció

Mivel az írásban többször előfordul ez a két fogalom, és nagyon gyakran összekeverik őket, jónak látjuk pontosítani, melyik mit jelent. Hangtanilag nagyon közel állnak egymáshoz, de jelentésükben komoly különbség van. A definíció a fordítóprogramnak szóló üzenet, amelyben közöljük egy általános értelemben vett objektum (például változó vagy függvény) jellemzőit. Tesszük ezt anélkül, hogy ilyen jellemzőjű változópéldányt létrehoznánk. A definíció „csupán" azért szükséges, hogy a fordítóprogram tisztában lehessen a definiált objektum mibenlétével. A deklaráció ezzel szemben konkrétan létrehozza a megjelölt objektumot (területet foglal a számára). A két dolog gyakran egybemosódik. Ilyenkor a definíció és a deklaráció egy forrásnyelvi egységben kap helyet. Nézzünk néhány példát az ANSI C nyelvterületen:

| | |
|---|---|
| DEFINÍCIÓ | extern int var; |
| | struct st {int v1, char v2}; |
| | int fnc(int v1, float v2); |
| DEKLARÁCIÓ | struct st myst; |
| | int fnc(int v1, float v2) {;} |
| EGYBEMOSVA | struct st {int v1, |
| | char v2} myst; |

A funkciódeklarációt besorolhattuk volna az egybemosva című fejezetbe is, mivel a definíciót itt meg kell ismételni, nem írhatom azt, hogy fnc {;}, még akkor sem, ha az fnc függvényt előzőleg már definiáltam.

## OOP szakszótár

Encapsulation — egységbezárás, az objektumosztály mezőinek hozzáférés-védelme (private, protected, public).
Polymorphism — többrétűség, a származtatott osztályok átdefiniálhatják az örökölt függvénymezőket (azonos név alatt létrehozhatják a függvény egy másik verzióját). Ehhez az őstípusban a függvénymezőnek virtuálisként kell definiálva lennie. Hogy az azonos nevű függvények közül mikor melyiket kell aktivizálni, ez attól függ, hogy melyik osztályból való objektum hivatkozik rá.
Inheritance — öröklés, az osztályok definiálásakor beépíthetünk már meglévő osztályokat. A beépített osztály az őstípus, amelyikbe beépítjük, az a származtatott típus. A származtatott típus örökli az őstípus valamennyi mezőjét, de az őstípusban privátként definiáltakra közvetlenül nem hivatkozhat.
Derived class — származtatott típus, olyan osztálydefiníció, amelybe beépítünk egy (vagy több) őstípust.
Base class — őstípus.
Late binding — késői összerendelés, az az eset, amikor csak futásidőben dől el, hogy az azonos nevű virtuális függvények közül az adott függvényhívás melyiket aktivizálja.
Over loading — korai összerendelés, a late binding esettel ellentétben a fordítási időben eldönthető, hogy melyik azonos nevű függvényre vonatkozik a függvényhívás.

tatott típusnak vagy származtatott osztálynak nevezzük. Az egymásba ágyazás mélysége korlátlan. Minden származtatott típus tartalmazza az összes őstípus mezőit (a függvénymezőket is). A mezőöröklés nemcsak a közvetlen ősök mezőire vonatkozik, hanem az ősök őseinek őseire is, a hierarchia teljes mélységében. Az egymásba ágyazásnak két módja lehetséges. Az egyik az, ahogy azt az ANSI C nyelvben megszoktuk, a származtatott osztály egyik mezője egy beágyazott osztály típusú objektum, vagy egy ilyenre mutató pointer. A másik már C++-specialitás, ez támogatja a C++ adatrejtési koncepcióját is. Elöljáróban gondolkodjunk el azon, hogy az ANSI C nyelvben egy — mondjuk — ötszörösen beskatulyázott struktúrataghoz hogyan férhetünk hozzá. Hát valahogy így:

```
skatulya5->skatulya4->skatulya3
->skatulya2->fontosadat
```

Nem túl jópofa, főleg, ha minden harmadik sorban kétszer kell hivatkozni rá.

Az adatrejtési koncepció lényegének megértéséhez képzeljünk el egy többszintű osztályhierarchiát. Van egy bázisosztályunk, valamint egy származtatott osztályunk. Minkét osztály tartalmaz adatmezőket és funkciómezőket is. Az osztálydefiníció során minden egyes mezőnek megszabhatjuk a hozzáférési szintjét (az adat- és a függvénymezőknek is). Háromféle szint közül válogathatunk. 1: public; az ilyen mezők hozzáférési szintje megegyezik az ANSI C struktúrák mezőijével. Érvényességi idejük alatt bárki, bármikor hozzájuk férhet. 2: private; az így definiált mezőket kizárólag az őket tartalmazó osztály funkciói láthatják. Minden külső hivatkozás hibajelzést eredményez. 3: protected; ez átmenet az előző két szint között. A protected mezők láthatóak az őket tartalmazó osztályban, valamint az összes ebből származtatott osztályban is, de a külső hivatkozások itt is hibajelzéshez vezetnek.

Egy másik fontos újítás az ANSI C-hez képest, hogy használhatunk több azonos nevű funkciót. Ezeknek vagy a paraméterszignatúrájukban kell különbözniük egymástól, vagy pedig másmás osztálynak kell tartalmaznia őket. Értelemszerűen a másodikként említett módszert csak osztálymezőként definiált funkciókra alkalmazhatjuk. A különböző paraméterszignatúrával rendelkező funkciókat az overloaded jelzővel illetjük. Azokat, amelyeknek még a paraméterszignatúrájuk is azonos, csak az őket tartalmazó osztályban van különbség, virtuálisként tartjuk számon.

**Fridl György**

# A C nyelv Unix-felülete IV.

# Még mindig olvasunk...

Ahogy ígértük, a múlt hónapban beleszagoltunk az innen-onnan, így és úgy olvasás territóriumába, de akkor csak éppenhogy. Most jön a teljes mintavétel.

A scanf, fscanf, sscanf utasításoknál a típusjelzés előtt meg lehet adni maximális mezőszélességet, aminek elsősorban string típusnál van jelentősége. Ha a mező hosszabb a megadott értéknél, akkor csak a specifikált számú bájtot olvassa be a program:

```
fp = fopen("DATA","r");
fscanf(fp,"%3s",buff);
printf("%s\n",buff);
```

DATA tartalma:

```
bbaccddABCD
```

Kimenet:

```
bba
```

A ritkábban használt integer típusok: u (unsigned decimális), o (oktális), x (hexadecimális). Például:

```
fp = fopen("DATA","r");
fscanf(fp,"%u%d",&x,&y);
printf("x=%d(d) x=%u(u)
y=%d(d) y=%u(u)\n",x,x,y,y);
```

DATA tartalma:

```
4294967295 -1
```

Kimenet:

```
x=-1(d) x=4294967295(u)
y=-1(d) y=4294967295(u)
```

Long, short és double argumentum specifikálására van lehetőség. A d, u, o, x konverziós karakterek elé írt l azt jelzi, hogy long integerre mutató pointer a megfelelő argumentum; a h betű ugyanitt short integert jelöl. Az e és f típus esetén az l betű double típust jelez.

### Számrendszerek közötti konverzió

A scanf függvények segítségével (és a printf-ekkel is) egyszerűen tudunk

egyik számrendszerbeli ábrázolásmódról egy másikra áttérni:

```
strcpy(buff1,"16");
sscanf(buff1,"%d",&x);
sprintf(buff2,"%o",x);
printf("%s\n",buff2);
```

Kimenet:

```
20
```

## A c, azaz karaktertípus

A karaktertípus különlegessége, hogy ennél nem lépi át a program a fehér karaktereket:

```
fp = fopen("DATA","r");
fscanf(fp,"%c%c",buff,buff
+ 1);
printf("buff[0]=%c
buff[0]=%d\n",buff[0],buff[0]
);
printf("buff[1]=%c\n",buff[1]
);
```

DATA tartalma:

```
 abcd
```

Kimenet:

```
buff[0]=  buff[0]=32
buff[1]=a
```

(Az első beolvasott karakter a szóköz volt, amelynek a kódja 32.) A fehér karakterek átugrását c típusnál is be lehet kapcsolni, ha a %c elé legalább egy szóközt teszünk:

```
fp = fopen("DATA","r");
fscanf(fp," %c
%c%c",buff,buff + 1,buff +
2);
printf("buff[0]=%c
buff[1]=%c",buff[0],buff[1]);

printf("buff[2]=%c\n
buff[2]=%d\n",buff[2],buff[2]
);
```

DATA tartalma:

```
 a b cd
```

Kimenet:

```
buff[0]=a buff[1]=b buff[2]=
buff[2]=32
```

## Karakterek közvetlen (literális) megadása

Ha a formátumvezérlő stringben közvetlenül adunk meg karaktereket, akkor ezek ugyan be lesznek olvasva, de nem tárolódnak el a beolvasott értékek. A bemeneten a megfelelő pozícióban pontosan a várt karaktereknek kell szerepelniük, ellenkező esetben a beolvasás itt befejeződik. Például:

```
fp = fopen("DATA","r");
fscanf(fp,"%dm2%dtonna",&x,&y
);
printf("x=%d y=%d\n",x,y);
```

DATA tartalma:

```
 469m2  139tonna
```

Kimenet:

```
x=469 y=139
```

Figyeljük meg, hogy az 'm2' és a 'tonna' karaktersorozatoknak közvetlenül a számok után kellett következniük! Ha azt akarjuk, hogy a számok után tetszőleges számú (nulla vagy több) fehér karakter is állhasson, akkor úgy kell eljárnunk, mint a karaktertípusnál: a literálisan megadott karaktersorozat elé legalább egy szóközt kell írnunk:

```
fp = fopen("DATA","r");
fscanf(fp,"%d m2%d
tonna",&x,&y);
printf("x=%d y=%d\n",x,y);
fscanf(fp,"%d m2%d
tonna",&x,&y);
printf("x=%d y=%d\n",x,y);
fscanf(fp,"%d m2%d
tonna",&x,&y);
printf("x=%d y=%d\n",x,y);
```

DATA tartalma:

```
469m2 139tonna
469 m2 139 tonna
469
m2 139 tonna
```

Kimenet:

```
x=469 y=139
x=469 y=139
x=469 y=139
```

Ha '%' karaktert akarunk megadni, akkor azt meg kell kettőzni.

## Tételek átlépése

A scanf, fscanf, sscanf utasításoknál van még egy érdekesség. Például a %-jel után elhelyezett * karakter hatására scanf beolvassa a megfelelő tételt, de nem teszi el a memóriába, tehát valójában átlépi:

```
fp = fopen("DATA","r");
fscanf(fp,"%*d%d",&x);
fscanf(fp,"%*3s%s",buff);
printf("x=%d
buff=%s\n",x,buff);
```

DATA tartalma:

```
123 678
12345678
```

Kimenet:

```
x=678 buff=45678
```

## A scanf visszatérő értéke

„Normális" esetben scanf a sikeresen beolvasott és memóriába töltött tételek számát adja vissza. Ha a formátumspecifikáció és az adatok között ellentmondást észlel, akkor a beolvasás befejeződik, a „hibás" tétel a bemeneten marad. (Ezt abból lehet észrevenni, hogy a scanf visszatérő értéke kisebb a vártnál.) Ha az első beolvasott karakter vagy észlelt ellentmondás előtt fájlvégbe ütközik, akkor EOF-ot ad vissza. (Azaz –1-et.) Ennek az lehet az oka, hogy nem tudtuk a fájlt megnyitni, nincs jogunk olvasni, esetleg például csak szóközök vannak benne, mi pedig egy számot vártunk. Például:

```
fp = fopen("DATA","r");
r  = fscanf(fp,"%%%d
%d",&x,&y);
printf("r=%d x=%d
y=%d\n",r,x,y);
```

DATA tartalma:

```
%123 768
```

Kimenet:

```
r=2 x=123 y=768
```

## A karakterosztály-típus

Ez a típus rendkívül hasznos, de szintaxisa nehezebben tanulható meg, mint a többié. A karakterosztály is egy fajtája a stringtípusnak. Szögletes zárójelek között megadhatunk karaktereket, a beolvasás mindaddig folytatódik, míg a felsoroltak közé tartozó karakterek következnek:

```
fp = fopen("DATA","r");
fscanf(fp,"%[abcd]",buff);
printf("buff=%s\n",buff);
```

DATA tartalma:

```
bbaccddABCD
```

Kimenet:

```
buff=bbaccdd
```

A karaktertípushoz hasonlóan itt sincs fehérkarakter-átlépés, kivéve, ha a %-jel elé legalább egy szóközt teszünk. Ha a soron következő karakter nem tartozik a megadottak közé, akkor a beolvasás befejeződik. A szögletes zárójelek között intervallumot is meg lehet adni:

```
fp = fopen("DATA","r");
fscanf(fp,"%[a-cAm-o]%d",buff
);
printf("buff=%s\n",buff);
```

DATA tartalma:

```
nnoAmbbaccddBCD123
```

Kimenet:

```
buff=nnoAmbbacc
```

Ha a nyitó szögletes zárójel után közvetlenül ^ jel van, akkor az összes olyan karakter fog megfelelni a specifikációnak, amely nincs felsorolva:

```
fp = fopen("DATA","r");
fscanf(fp,"%[^0-9]",buff);
printf("buff=%s\n",buff);
```

DATA tartalma:

```
abcde1234
```

Kimenet:

```
buff=abcde
```

Ezt a fajta beolvasást például ott tudjuk igen jól hasznosítani, ahol adott mezőszeparátorokkal elválasztott mezőkből álló rekordokat akarunk olvasni. Tegyük fel például, hogy '#' jelekkel van határolva 3 mező minden rekord-

ban, a rekordokat soremelés karakterek választják el egymástól, és be akarjuk olvasni egy rekord mezőit a buff1, buff2, buff3 tömbökbe:

```
fp = fopen("DATA","r");
fscanf(fp,"%[^#]#%[^#]#%[^#]#
\n",buff1,buff2,buff3);
printf("buff1=%s buff2=%s
buff3=%s\n",buff1,buff2,buff3
);
fscanf(fp,"%[^#]#%[^#]#%[^#]#
\n",buff1,buff2,buff3);
printf("buff1=%s buff2=%s
buff3=%s\n",buff1,buff2,buff3
);
```

DATA tartalma:

```
123#768#aa bb cc#
123#768#aa bb cc#
```

Kimenet:

```
buff1=123 buff2=768
buff3=aa bb cc
buff1=123 buff2=768
buff3=aa bb cc
```

A '#' és a soremelés karakter közé azért írtunk egy szóközt, hogy amennyiben a '#' után esetleg még lennének fehér karakterek, azokat lépje át a program.) A karakterosztály elemeként a 'speciális' karakterek (\t, \n, \v, \ddd) is megadhatók:

```
[ab\nc\t]  [^\61]
```

A karakterosztály-típust mezőszélesség-megadással kombinálva arra lehet használni, hogy adott hosszúságú részt (mezőt) olvassunk egy rekordból. A szögletes zárójelek között olyan karaktert kell megadni, amely biztosan nem fordul elő a rekordban, például a rekordszeparátort. Tegyük fel, hogy a rekordokat soremelés karakterek választják el egymástól, és a 6. pozícióban kezdődő 10 karakteres mezőt szeretnénk beolvasni:

```
fp = fopen("DATA","r");
fscanf(fp,"%*5[^\n]%10[^\n]",
buff);
printf("buff=%s\n",buff);
```

DATA tartalma:

```
abcde123 567 90ABCDEF
```

Kimenet:

```
123 567 90
```

Érdemes megjegyezni, hogy karakterosztály a reguláris kifejezésekkel dolgozó programoknak (sed, grep, awk, expr) is megadható, de a szintaktika egy kicsit más: a szögletes zárójelek között specifikált karakterek ott egyetlen karaktert írnak le, nem egy karaktersorozatot.

Egy másik különbség az, hogy a scanf-fel ellentétben ott a szögletes zárójelek között nincs escape karakter, nem lehet például a tabulátort a \t karakterpárral leírni.

**Nemes Mihály**

## Hogyan készül egy Snobol program?

# Csak könnyedén, könnyeden...

Még egy új szövegszerkesztő megtanulására sem egykönnyen szánja rá magát egy programozó, nemhogy egy új programnyelvére.
Legfeljebb akkor, ha különleges fantáziát lát benne.
Kérjük olvasóinkat, halasszák el döntésüket, és ne megtanulni akarják azt, amiről szó lesz, csak megfigyelni. Esetleg gondolkozzanak el rajta...

Induljunk ki a következő feladatból. Valamelyik szöveges állományunkról meg akarjuk tudni, hogy hány szó van benne, és az előforduló szavak hogyan oszlanak meg hosszúság szerint. Hogy nehezebb legyen a feladat, engedjük meg a szavakban a magyar ékezetes betűket is.

Vegyük sorra, milyen lépésekre van szükségünk, és milyen eszközökkel rendelkezik ezek megfogalmazásához a Snobol nyelv!

### 1. probléma. A karakterkészlet

Szükségünk lesz a magyar nagybetűk és kisbetűk felsorolására. Válasszunk ki egy változót a nagybetűk tárolására, legyen ennek neve például hucase (Hungarian upper case). Ebbe be kell vinnünk az angol ábécé nagybetűit — ez már készen megtalálható a rendszer &ucase kulcsszóval megjelölt „védett" változójában, és ki kell egészítenünk az ékezetes betűkkel. A megfelelő utasítás:
hucase = &ucase 'ÁÉÍÓÖŐÚÜŰ'

Mint látható, az értékadás jele az egyenlőségjel. Jobboldalt az idézőjel nélkül megadott változót az utasítás kiértékeli, majd a kiértékelésből kapott füzér végéhez „hozzáragasztja" az idézőjelben megadott füzért. A füzérek egymáshoz ragasztásának, az ún. „konkatenáció" műveletének egyszerűen a szóköz a jele.

Tökéletesen az előző utasítás mintájára végezhetjük el a kisbetűk tárolását, mondjuk a hlcase (Hungarian lower case) változóban:

```
hlcase = &lcase 'áéíóöőúüű'
```

Itt jegyezzük meg, hogy a kisbetűk-nagybetűk megkülönböztetését a rendszer csak az idézőjelek között megadott karaktersorozatoknál végzi el következetesen minden esetben. Mint később látni fogjuk, a címkéknél, változóknál, függvényeknél stb. nem, hacsak külön nem kérjük. Így például a HLCASE és a hlcase ugyanarra a változóra utalna.

További megjegyzés: vigyázzunk a szóközzel a Snobolban! Nemcsak hogy nem „átlátszó" karakter a Snobolban, hanem sok helyen nagyon fontos szerepet tölt be. Így például sokaknak szokatlan lesz, hogy a bináris operátorok mindkét oldalán kötelezően szóköznek kell állnia. Figyeljük meg, hogy még az egyenlőségjel két oldalán is mindig ott a szóköz!

### 2. probléma. A szavak leválasztása

Más. A sor első pozíciójának a Snobolban kitüntetett szerepe van. Ha például címkéje van az utasításnak, annak föltétlenül itt kell kezdődnie, változó viszont nem kezdődhet az első pozícióban.

Szükségünk lesz egy olyan mintaillesztő apparátusra, amelynek segítségé-

vel fel tudjuk ismerni és le tudjuk választani a bejövő karakterek folyamatából a következő szót. Külön probléma a mintaillesztés, és külön-külön részprobléma a szó kezdetének és a szó végének a felismerése.

## 2.1 alprobléma. Hol kezdődik a szó?

A szó kezdetének felismerése annyit jelent, hogy keresztül kell engednünk egy „szűrőn" mindazokat az egymás utáni jeleket, amelyek nem tartoznak bele egy meghatározott jelkészletbe — esetünkben a kisbetűk és nagybetűk halmazába.

## 2.2 alprobléma. És hol végződik?

A szó végének felismerése éppen az ellenkező tevékenységet kívánja meg: a szűrőnek mindaddig át kell engednie a jeleket, míg „idegen" jellel nem találkozik. Az adott esetben a „kebelbeli" jelek a kisbetűk és nagybetűk, az „idegenek" az összes többiek.

Ezt a két alapvetően fontos feladatot két beépített függvényre bízhatjuk. Az első feladat elvégzésére szolgál a BREAK elnevezésű függvény, a másikra a SPAN. A BREAK pontosan az argumentumában megadott jelek előtt engedi le a sorompót, a SPAN pedig azok után. Argumentumaikkal együtt a két függvényt tehát így adhatjuk meg:

break(hucase hlcase)
span(hucase hlcase)

Fel is sorolhatnánk a jeleket (persze akkor idézőjelek közé zárva) — de minek, ha a változók nevével könnyebben és egyszerűbben hivatkozhatunk rájuk.

Ha valakinek tetszik, össze is vonhatja egy változóba a kis- és nagybetűket például ezzel az utasítással:

betuk = hucase hlcase

Ebben az esetben még egyszerűbben, break(betuk) és span(betuk) alakban kellene aktivizálnunk a fenti „betűkereső" és „betűtaláló" függvényeinket.

## 2.3 alprobléma. „Karanténba zárva"

A sorompó leengedése csak egy része annak a tevékenységnek, amit végre kell hajtanunk. A sorompón átjutottakat szükség esetén együtt kell tartanunk, ugyanabban a sorrendben, ahogy jöttek. Bizonyos értelemben ehhez is egy értékadó műveletet kell tudnunk elvégezni, de menet közben, a mintaillesztés részeként. Nevezzük ezt, mondjuk, „karantén-értékadásnak". Jele: egy pont (előtte, utána szóköz), majd a változó neve.

A fentiek ismeretében már könnyen el tudunk készíteni a mintaillesztéshez egy összetett mintát is:

```
szokep = break(hucase
hlcase) span(hucase
hlcase) . szo
```

Vagy áttekinthetőbben, a kis- és nagybetűk összevonásával:

```
szokep = break(betuk)
span(betuk) . szo
```

Megtehetnénk, hogy a BREAK függvény után is beillesztünk egy „karantén-értékadást", ha kíváncsiak vagyunk, hogy milyen karaktersorozatokat kellett átengednünk két szó között:

```
szokep = break(betuk) . br
span(betuk) . szo
```

Figyeljük meg a „pontoperátor" és a „szóközoperátor" prioritási viszonyát! A „pontoperátor" erősebben köt, mint a „szóközoperátor", vagyis azt előbb kell végrehajtani. A SZO változóba így csak a SPAN függvény eredményét lehet majd bevitetni ennek a mintaillesztő változónak a segítségével.

Zárójelekkel persze másképp is intézkedhetnénk erőviszonyaik megváltoztatására. Például az f és g függvényekre, ha azt akarjuk, hogy előbb hajtódjék majd végre a szóköz által előírt „összeragasztás" művelete, és csak azután a „karanténba zárás", akkor így kell felkészítenünk a mintaillesztő változót:

```
minta = (f(a) g(b)) . m
```

## 2.4 alprobléma. Hogy lesz ebből mintaillesztés?

A mintaillesztő változóval már megvagyunk, de még el kell végeztetnünk magát a mintaillesztést. Ehhez partner is kell: egy olyan karaktersorozat, amelynél van mit keresnünk a már felajzott mintaillesztő változó értékének segítségével.

Tegyük fel, hogy a következő sor tartalma már belekerült a SOR elnevezésű változóba. Mivel a minta már készen van a SZOKEP változóban, csupán ennyit kell írnunk:

```
sor szokep
```

A SZOKEP minta fenti két változatának akármelyikét használjuk, ennek az utasításnak a hatására a rendszer a SOR elnevezésű változóra alkalmazza a SZOKEP elnevezésű mintát. Ez viszont már azt is magában foglalja, hogy a SZO „karantén változóban" megjelenik a következő szó.

Mit jelent az, hogy „következő"? Úgy képzelhetjük, hogy a rendszernek van egy láthatatlan „mintaillesztő kurzora", amely balról jobbra mozog, de nem magukra a karakterekre mutat, hanem — bármilyen furcsa — mindig két karakter közé. A mintaillesztés előtt a kurzor a SOR változóval jelölt karaktersorozat első karaktere elé mutat. Ha a fenti mintaillesztő utasítást az első esetben alkalmazzuk, a BREAK függvény hatására továbbmozog a kurzor, de hirtelen lefékez az első szó első betűje előtt. (Ha az első szó rögtön a sor elején kezdődik, akkor persze nulla hosszúságú távolságot halad előre a kurzor.)

Tegyük fel, hogy a BR változó „él", vagyis a SZOKEP minta második változatát alkalmazzuk. Ekkor a BR változóba bekerül a kurzor által végigsöpört karaktersorozat — az adott esetben az első szó előtt álló nem betű karakterek sorozata. A mintaillesztés azonban ezzel nem ér véget, hanem folytatódik a SPAN függvénnyel. Ennek hatására a kurzor továbbmegy a szó utolsó betűje utánig, a SZO változóba pedig beíródik az első szó.

Ha újra alkalmazzuk ugyanezt az utasítást, akkor a „mintaillesztő kurzor" onnan indult, ahová előző alkalommal került, az utasítás tehát a második szót fogja beírni a SZO változóba.

## 3. probléma. Cikluskészítés

Nyilvánvalóan szükségünk lesz egy ciklus kiépítésére, amely addig tart, míg a sorból el nem fogynak a szavak, és egy másik ciklusra, amely valamilyen titokzatos inputból újratölti a SOR változót.

A ciklus felépítéséhez ügyes segítséget nyújt a szelektív ugróutasítás, amely különbséget tesz sikerességük alapján az utasítások között. Ha sikeres volt az utasítás végrehajtása (s, vagyis success), akkor máshová tud ugratni, mint sikertelen utasítás végrehajtás után (f, azaz failure).

Körülbelül így képzelhetjük el globálisan a sztenderd Snobol utasítások szerkezetét:

címke tennivaló folytatás
(opcionális) (opcionális)

Mint látjuk, a megfelelő folytatásra az utasítássor végén utalhatunk. Fontos azonban a folytatás megadási módja is. Függvényszerű jelöléssel utalhatunk rá, hogy sikeres vagy sikertelen esetben, vagy esetleg mindkettőben végrendelkezni akarunk-e. Végül van még egy lehetőség: ha feltétel nélkül másutt kérjük a program folytatását. Szemrevételezzük a teljes árukínálatot:

```
:s(címke1)
```

```
:f(címke2)
:s(címke3)f(címke4)
:(címke5)
```

A kettőspont nem nyomdahiba! Ez választja el a folytatáskijelölő részt (ha van ilyen) az utasítás előző részétől. A szóközre itt is ügyelnünk kell: a kettőspont előtt legalább egy szóköz kötelező, utána viszont nem kell.

Mi történik abban az esetben, amelyikről nem rendelkezünk külön? Ilyen esetben a vezérlés átadódik a következő utasítássornak.

Most már ki tudjuk dekorálni címkékkel mintaillesztő utasítássorunkat. Legyen a sor „neve" (címkéje) KOVSZO, és sikertelen mintaillesztés esetén kérjük a folytatást az OLVAS címkéjű utasítástól:

```
kovszo   or szokep =: f(olvas)
```

Újabb ravaszság is van azért ebben az utasításban. Mivel az értékadó utasítás jobb oldalán az üresszó áll, a feldolgozott szó helyére az üresszó íródik be a SOR változóban. Ez azonban nem baj, mert a fifikusan elkészített SZOKEP mintaillesztő változó a mintaillesztés során elteszi a felismert szót a SZO változóba. Mint már tudjuk, sikeres végrehajtás után a rendszer továbblép a következő utasítássorra, ha azonban a mintaillesztés sikertelennek bizonyulna, akkor az OLVAS címkénél folytatódik a program végrehajtása.

## 4. probléma. A bemenet

Alakítsuk ki a bemenetet. Az adott esetben az a legegyszerűbb, ha a program parancssorában adjuk meg átirányítással a feldolgozandó fájl nevét. Legyen a Snobol programunk neve SZOHOSSZ.SNO. Végrehajtásához meg kell hívnunk a SNO.EXE programot.

Adatokként irányítsuk át hozzá egy szövegfájlunk tartalmát:

```
C>sno szohossz.sno
< myinput.txt
```

Ha már ismeri a program a bemeneti fájl nevét, magában a programban elég megadnunk az OLVAS címkéjű utasítássort:

```
olvas sor = input :f(kesz)
```

Ennek hatására a bemeneti fájlból soronként bekerülnek az információk a SOR elnevezésű változóba. Az utasítássor mindaddig sikeresen végződik (vagyis a program a következő utasítássorral folytatódik), míg az EOF jelhez nem jut a feldolgozás.

Az EOF hatására a sikertelen esetre megadott rendelkezésünk lép életbe: a végrehajtás a KESZ címkénél folytatódik.

## 5. probléma. A hossz szerinti számlálás

Gondoskodnunk kell egyrészt a szavak hosszának megállapításáról, másrészt a hossznak megfelelően egy-egy számlálórekesz kialakításáról. A hossz megállapítását elvégzi a SIZE függvény, a számláláshoz pedig bevezetünk egy egydimenziós tömböt SZAML néven. Ennek indexeléséhez közvetlenül felhasználhatjuk a SIZE függvényből kapott értékeket.

A tömb bevezetése csupán ennyiből áll:

```
szaml = array('30')
```

Az idézőjel nem hiba, bár egydimenziós tömb esetén az elhagyása sem lenne az. Érdemesebb azonban az általános esetre felkészülve azt mondani, hogy füzérként kell megadni a tömb dimenzióját és méreteit. A '30' itt a tömb egyetlen dimenziójának méretét jelöli, '10,20' egy 10x20-as méretű kétdimenziós tömböt jelölne, '10,20,30' egy 10x20x30-as háromdimenziós tömböt.

A tömb elemei a SZAML<1>, SZAML<2>, ..., SZAML<30> névre hallgatnak. Kétdimenziós tömb elemére utalhatnánk például a T,4 jelöléssel — persze csak akkor, ha előzőleg definiáltunk volna egy T<3,4> tömböt például a következőképpen:

```
t = array('6,10')
```

Az tömb indexét tetszőleges kiértékelhető függvénnyel megadhatjuk. Ha például a szavak hosszával akarjuk indexelni a SZAML tömböt, akkor a tömb elemeire mint változókra a SZAML<SIZE(SZO)> jelöléssel hivatkozhatunk.

## 6. probléma. A kimenet

Kérhetjük a kimenetet a képernyőre vagy valamilyen fájlba. Mindig a képernyőre íródik ki a SCREEN változónak átadott érték. Az OUTPUT változó értéke akkor kerül a képernyőre, ha nem intézkedtünk róla másképp. Legegyszerűbben úgy írathatjuk fájlba az OUTPUT változóba tett értékeket, hogy a parancssorból fájlba irányítjuk a standard outputot, például:

```
C>sno szohossz.sno
< myinput.txt > myoutput.txt
```

## 7. probléma. Feltételes értékadás

Nagyon tömören lehet megfogalmazni a feltételes értékadást olyan beépített függvények segítségével, amelyek „kámfort játszanak", vagyis üresszót állítanak elő, ha a feltétel teljesül. Nézzük például a következő értékadást! Itt az i változó értéke akkor és csak akkor nő i + 1 -re, ha i jelenlegi értéke kisebb, mint 30 (lt = less than). Ha nagyobb vagy egyenlő, akkor az értékadás végrehajtása félbeszakad, és a vezérlés átadódik az ERNYO címkéjű utasításnak.

```
i = lt(i,30) i + 1
:f(ernyo)
```

Ha ugyanis a feltétel teljesül, akkor az történik, hogy a keletkező üresszót a rendszer egyesíteni akarja az utána következő kifejezéssel. Az üresszóval való egyesítés azonban változatlanul hagyja a másiknak az értékét — nemcsak füzérek esetében, hanem számértékkel bíró változók esetében is.

## 8. (és majdnem utolsó) probléma. Változók értéke futás közben

A program belövésekor felbecsülhetetlen segítséget nyújt a jó nyomkövetés. Kommentárok formájában (az első pozícióban *-ot tartalmazó álutasításokként) beépítettük a TRACE függvényt a programba. Ha itt kitöröljük hol az egyik, hol a másik csillagot, akkor a TRACE függvény aktivizálódik, és az adott változó értéke minden változáskor bekerül az OUTPUT-ba. (A TRACE függvény egyéb lehetőségeiről majd később ejtünk szót — egyelőre elég, ha így ismerkedünk vele.)

A teljes programot már előző számunk lemezmellékletén közreadtuk. Viszont csak most jutottunk el oda, hogy majdnem mindent ismerünk is már hozzá.

Talán csak a felhasznált beépített függvények kívánnak némi magyarázatot. A TIME() függvénnyel információt kaphatunk a futás kezdetétől eltelt időről. Azért vettük bele a programba, hogy ne legyen üres a képernyő addig sem, amíg a program dolgozik. (Ki tudja, milyen hosszú fájlon próbálja ki a kedves olvasó a programot...) Az NE a NOT EQUAL rövidítése. Második argumentumát elhagyhatjuk, ha nullához akarunk egy számot hasonlítani — itt ez történik. Az LPAD függvénnyel tabuláltuk a változók kiíratását, az üres helyeket ez balról szóközökkel tölti fel („left padding").

Jó szórakozást a program kipróbálásához azoknak is, akik az előző szám „no comment" tálalása után csak most mernek hozzányúlni. Ne mulasszák el módosítgatni legalább a csillagos sorokat. Nézzék meg például, mi történik, ha a &trace kulcsszónak különböző értékeket adunk, mondjuk 6-ot, 0-t, —1-et.

**Vargha Dénes**

INFORMÁCIÓKÉRÉS: 69 ▶



INFORMÁCIÓKÉRÉS: 70 ▶



INFORMÁCIÓKÉRÉS: 71 ◀

# Hardver-szoftver — innen-onnan

E havi palettánkat olyan hardver-szoftver színekkel „tarkítjuk", amelyek egymástól meglehetősen távol eső színvilágot nyitnak meg előttünk. A bemutatandó termékeket — digitalizáló táblát, szervercsaládot, irodai automatizáló, illetve utazásszervezési szoftvert — a kényelmes használat és a megbízható működés mellett a viszonylag alacsony ár is jellemzi.

## Menet közben kivehető

Közel egy éve számoltunk be arról, hogy a magyar piacon megjelent egy olyan RISC-alapú Unix gépcsalád (Data General AViiON), amely kedvező ár/teljesítmény mutatóival kiemelkedik a hasonló kategóriájú hazai hardverkínálatból.

Az azóta is sikeresnek bizonyuló gépcsalád „felső végén" most új tag jelentkezett.

Az új 8 processzoros gép (AV 6280 és AV 8000-8) a Motorola 88100 RISC processzorára épül és teljesen szimmetrikus vezérlést biztosít, 235 VAX-MIPS teljesítményt nyújt, 128-768 MB operatív tárat és 5 MB cache memóriát tartalmaz.



A gépcsalád „alsó végén" is történtek fejlesztések. Így az AV 4000-es sorozat új gépeihez (AV 46x5) nagy megbízhatóságú háttértároló alrendszereket csatlakoztattak. A diszk alrendszer úgy osztja el 5 lemezen az információkat, hogy minden 4 egységnyi tárolt adathoz tartozzon egy egységnyi redundáns adat. Így bármelyik lemez esetleges kiesésekor a rendszer nem veszi észre a meghibásodást. A rossz lemezt akár üzemszerű működés közben is kicserélhetjük, úgy, hogy a rendszerben dolgozó felhasználók ebből semmit sem vesznek észre.

Ezek a diszk alrendszerek már illeszthetők a Microsystem Rt. által forgalmazott AViiON 4000-es sorozathoz. Így olyan nagy teljesítményű és nagy megbízhatóságú, közép-mini kategóriájú számítógép jelent meg a hazai piacon, amely 7 millió forint körüli árával maximálisan kielégíti a központi online információs szerverrel szemben támasztható követelményeket.

## Mint a denevér...

A hagyományos digitalizáló táblákkal ellentétben egy olyan amerikai eredetű digitalizáló került hazai forgalomba, amely nemcsak lényegesen olcsóbb, hanem jóval kényelmesebben is használható elődeinél.

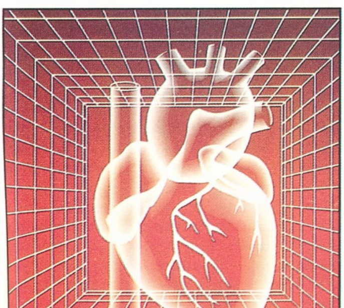
E digitalizáló érdekessége, hogy nem tartozik hozzá tábla. A hagyományos digitalizálókkal ellentétben — ahol a rajzot tesszük fel berendezésre — itt magát a szerkezetet helyezzük tetszőleges felületre (röntgenképre, filmre, földre, ablakra, térképre...). A 3 dimenziós változat 2,5x2,5x2,5 méteres mérethatáron belül digitalizálhat, illetve követhet nyomon egy mozgássorozatot. A térbeli adatokat szolgáltató berendezés új távlatokat nyit meg előttünk, hiszen eddig térfogatok digitalizálására nem volt módunk.

A digitalizáló működési elve hasonló ahhoz, ahogy a denevérek tájékozódnak. A síkbeli ultrahangos szerkezetbe 2, a térbelibe 4 mikrofont helyeztek. Maga a digitalizáló (a ceruza, illetve a 4 gombos szálkereszt) bocsát ki hangjelet, a távolságot pedig együtt méri be a mikrofon és a szerkezet. A digitalizáló további érdekessége, hogy egyedül ennek a fajtának van programozható kimenete. Az adatokat akár ASCII, akár bináris alakban megkaphatjuk, és utólag bármilyen szoftverhez illeszthetjük. A szerkezeten található nyomógombok közvetlenül programozhatók: itt mondjuk meg, hogy mit akarunk digitalizálni (kezdőpontot, vonalat, ívet, kontúrt). A digitalizáló illeszthető ismert CAD-rendszerekhez, így közvetlen driverek biztosítják a kapcsolatot



például az AutoCAD, CadKey, Design Cad 3D vagy Versa-CAD rendszerekhez. A 3D-s változattal egyszerre akár 16 ember is dolgozhat, segítségével maximum 16 helyről származó információ vehető fel.

Az újfajta berendezést széles körben alkalmazhatjuk. Irányítástechnikai feladatok (vasúti térképek), sport (mozgásanalízis), egészségügy stb. Segítségével előállíthatjuk például a szív vagy a koponya háromdimenziós képét (idáig ultrahanggal csak síkbeli képet tudtak reprodukálni). A leggyakoribbak talán a térképészeti alkalmazások lesznek. Bármilyen bonyolult alkatrész térbeli méreteit digitalizálhatjuk (gépészet), de videoanimációra és robotvezérlésre is használható a berendezés. Építészeti, belsőépítészeti és látványtervezési feladatokban a direktbe forgatásoknál és a nézeti képek előállításánál különösen fontos szerephez juthat a szerkezet.



A digitalizáló berendezés gyors, 100 pont/s adatátviteli sebességgel működik. A forgalmazó PAB Kft. meglepően olcsón értékesíti a lapos, hordozható, kevés helyet elfoglaló, akár a földre is lerakható szerkezetet. Feleannyiba kerül, mint hagyományos társai, így például egy „közepes" digitalizáló (90x120 cm) 196 000 Ft a hozzá tartozó összes szoftverrel együtt. (Az igazi újdonságnak számító 3D-s digitalizáló pedig 1 245 000 Ft.) A digitalizáló üzembe helyezéséhez semmiféle hardverbeállítás nem szükséges, csak rá kell kapcsolni a számítógépünkre, amely lehet tetszőleges IBM PC, de munkaállomás is.

# A pult túlsó oldalán...

Az utazási irodák nyári programajánlatai között böngészve örömmel láttuk, hogy nemcsak az utazni vágyók gyorsabb és pontosabb informálására használnak már számítógépet, hanem az információs pult másik oldalán álló ügyintézők az utaztatással kapcsolatos valamennyi belső feladat nyilvántartását, adminisztrálását is számítógéppel végzik. Egy ilyen programcsomagot mutatunk most be, amely komplex szolgáltatásaival hatékonyan támogatja az utaztatási irodák tevékenységét.

Az Informixben írt, moduláris felépítésű alkalmazás olyan komplex rendszer, amely kezeli a ki- és beutazásokat, forgatókönyvet készít az idegenvezetők számára, továbbá ellátja a főkönyvi feladatokat is.

Egy utazási iroda általában sok partnerrel áll kapcsolatban, akik 1 évre előre jelzik egyéni és csoportos igényeiket. Ennek alapján szervezi meg az utazást, az étkeztetést és az egyéb programokat. Nyilvántartja az árualapokat (busz, étterem, szállás stb.), és a szállítókkal való szerződéseket (időpontok). Amikor az utazási irodánál helyeket foglalnak, akkor a

program előkalkulációt készít, legyártja a szállítóknak a megrendelőket és a partnereknek a visszaigazolásokat, melyekben egyúttal valamennyi várható költség és a várható eredmény mértéke is szerepel.

Az utazás után érkeznek a szállítói számlák az irodába. A program ezek alapján, és az előkalkulációk figyelembevételével állítja ki a vevői számlákat, amelyek a szoftver pénzügyi moduljába kerülnek. Itt jegyzi a program a vevői számlakiegyenlítéseket, kezeli továbbá a szállítói számlákat. Ezenkívül a modul vevői és szállítói analitikát is készít, és támogatja a főkönyvi feladásokat.

Tekintettel arra, hogy a vevői és szállítói pénzforgalmon kívül minden úthoz járulhatnak egyéb, előre nem látható költségek is, így az utazás végén utókalkulációs (vevői, szállítói, egyéb) táblázat is készül. Ez összehasonlítható az előkalkulációval, és egyértelműen kimutatható minden út eredménye.

Az utazási szoftverhez tartozik egy ügyes kis modul (forgatókönyv), amely jelentősen segíti az idegenvezetők munkáját. Az előzetes foglalások alapján állítják össze azokat a szöveges információkat, amelyekre az idegenvezető támaszkodhat (utasnévsor, városnézési terv, elszállásolás ütemezése stb.).

A szoftver telepítéséhez nem szükséges a teljes Informix, elegendő annak csak a futtató modulját megvásárolni. A programcsomag minden iroda sajátos igényeinek megfelelően telepíthető, a különféle listák rugalmasan állíthatók elő. A forgalmazó Areco Kft. a menet közben felmerülő változtatási igényeket rendszeresen nyomon követi, de a prompt kívánságokat is teljesíti. A szoftver tudásához képest méltányos áron, 149 000 Ft-ért kerül forgalomba, a fizetővendégszolgálati modul pedig külön 98 000 forint. A szoftverhez legalább 386-os gép szükséges (és 4 MB RAM, 80 MB winchester, monokróm monitor), Xenix vagy Unix operációs rendszer alatt. Aszinkron vonalon egy 486-os kártyával akár nagyobb utazási irodák igényei is kielégíthetők.



# Jót — olcsón

Az irodák világában dolgozók munkáját segíti az a nemrég Magyarországon is megjelent automatizálási szoftver, amely jóval hatékonyabbá teszi az irodai szervezeteket. A szoftver alkalmas jelentések készítésére, munkák ütemezésére, határidők figyelésére,üzleti ajánlatok összeállítására, levelek és körlevelek megírására, faxok és telexek küldésére, valamint a vállalati működés szervezésére.

Teljes mértékben magyarítva (magyar dokumentációval is!) kerül a hazai felhasználókhoz a Németországban az irodai automatizálás piacán „number one"-nak számító Prisma Office. A szoftver alkalmazkodik a felhasználó tudásszintjéhez, ugyanis lehetőséget ad rá, hogy akár kezdő, akár haladó, akár szakértői szinten kezeljük a rendszert, amelyre a könnyen érthető utasítások, a legördülő menük és a gyorsbillentyűk használata jellemző. A felhasználóktól nem idegen a szoftver saját fájlkezelő rendszerének használata sem, amely úgy működik, mint egy megszokott iratrendező szekrény.

A Prisma Office tartalmaz továbbá egy jól kezelhető szövegszerkesztő programot, és egy 100 000 szavas szótárat is. Azok a felhasználók, akik egész nap dolgoznak a rendszerrel, hatékony szövegszerkesztőre találnak benne, s a képernyőn egyszerre négy dokumentummal is dolgozhatnak. A felső szintű vezetők leginkább a különféle döntésekhez szükséges információk megkeresésére, jelentések áttekintésére használhatják. A rendszernek ugyan van saját beépített relációs adatbázis-kezelője, de megengedi, hogy más adatbázisokban (dBase, Paradox, Informix, Dataflex stb.), illetve táblázatkezelőkben vagy grafikai programokban tárolt információkhoz is hozzáférjünk.

A Prisma Office széles körű hazai elterjedését segítheti, hogy a szoftver a legtöbb számítógépen (IBM PC, Data General, hordozható gépek) és valamennyi jelentősebb operációs rendszer (DOS, OS/2, SCO-Unix, Unix) alatt futtatható. Tovább bővíti a rendszer alkalmazhatóságát, hogy nemrég készült el a Windowsos változat. Tekintettel arra, hogy egy szervezet különböző egységeiben dolgozó emberek munkáját segíti a Prisma Office, így elsősorban hálózatban érdemes használni. De van természetesen egy munkahelyes változata is. S ha hozzátesszük, hogy a rendszert forgalmazó Onyx Kft. kifejezetten kedvező áron értékesíti a szoftvert (egyedi verzió 55 000 Ft, 3 munkahelyes változat 119 000 Ft), akkor valóban nagy népszerűségre tehet szert ez a kényelmesen használható, magyarul beszélő, az irodai munkát automatizáló rendszer.

**Sziebig Andrea**




- Megbízható cég
- Legnagyobb választék
- Legnagyobb üzemeltetési tapasztalat

- Telefon alközpontok
- Hálózatok
- Időjelző rendszerek
- Irodatechnika

- Tervezés
- Tanácsadás
- Kivitelezés
- Üzemeltetés

Ingyenes
távközlési
és információs
tanácsadás

OKÉ!

COMEX

**BUDAPESTI TELEFON ALKÖZPONTI Kft.**

Budapest X., Bihari út 6. Tel: 127-7820
Levélcím: 1476 Pf. 176 Fax: 138-4079